UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
WACO DIVISION

COMMSTECH LLC, Plaintiff, v. CISCO SYSTEMS, INC., Defendant.	Case No. 6:19-cv-296 **COMPLAINT FOR PATENT INFRINGEMENT** **JURY TRIAL DEMANDED**

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Commstech LLC ("Commstech" or "Plaintiff") hereby asserts the following claims for patent infringement against Defendant Cisco Systems, Inc. ("Cisco" or "Defendant"), and alleges as follows:

SUMMARY

1. Commstech owns United States Patent Nos. 6,349,340; 6,606,317; 7,126,946; 7,152,231; 7,769,028; and 7,990,860 (collectively, the "Patents-in-Suit").

2. Cisco infringes the Patents-in-Suit by implementing, without authorization, Commstech's proprietary technologies in a number of its commercial networking products and related software including, *inter alia*, products that operate with the Cisco Internetwork Operating System ("IOS") XR software (including but not limited to Cisco CRS series routers, XR 12000 series routers, and ASR 9000 series routers), products that support the RFC 4607 specification related to "Source-Specific Multicast for IP" (including but not limited to products that operate with the Cisco NX-OS software, such as the Cisco Nexus series switches, Cisco MDS 9000 Family storage switches, and Cisco UCS 6100 Series Fabric

Interconnects), products that operate with the Cisco IOS software and comprise Content Addressable Memory ("CAM") and/or Ternary Content Addressable Memory ("TCAM") (including but not limited to Cisco Catalyst series switches that comprise a Switching Supervisor Engine), and products that operate with the "EasyQoS" application, which supports numerous Cisco routers, switches, and/or platforms listed on Cisco's website (https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-5-x/easyqos/supported-platforms/b_EasyQoS_Supported_Devices_1_5_x.pdf) (collectively referred to herein as the "Accused Products"). These Cisco products are marketed, offered and distributed throughout the United States, including in this District.

3. By this action, Commstech seeks to obtain compensation for the harm Commstech has suffered as a result of Cisco's infringement of the Patents-in-Suit.

NATURE OF THE ACTION

4. This is a civil action for patent infringement arising under the patent laws of the United States, 35 U.S.C. § 1 *et seq*.

5. Cisco has infringed and continues to infringe, and at least as early as the filing and/or service of this Complaint, has induced and continues to induce infringement of, and has contributed to and continues to contribute to infringement of, at least one or more claims of Commstech's Patents-in-Suit at least by making, using, selling, and/or offering to sell its products and services in the United States, including in this District.

6. Commstech is the legal owner by assignment of the Patents-in-Suit, which were duly and legally issued by the United States Patent and Trademark Office ("USPTO"). Commstech seeks monetary damages for Cisco's infringement of the Patents-in-Suit.

THE PARTIES

7. Plaintiff Commstech LLC is a Texas limited liability company with its principal place of business at 1708 Harrington Dr., Plano, Texas 75075. Commstech is the owner of intellectual property rights at issue in this action.

8. On information and belief, Defendant Cisco Systems, Inc. is a California corporation with a principal place of business at 170 West Tasman Dr., San Jose, California 95134. On information and belief, Cisco maintains at least two offices in this District at 12515 Research Blvd., Building 3, Austin, TX 78759, and 18615 Tuscany Stone, San Antonio, TX 78258.

9. On information and belief, Cisco directly and/or indirectly develops, designs, manufactures, distributes, markets, offers to sell and/or sells infringing products and services in the United States, including in the Western District of Texas, and otherwise directs infringing activities to this District in connection with its products and services.

JURISDICTION AND VENUE

10. As this is a civil action for patent infringement arising under the patent laws of the United States, 35 U.S.C. § 1 *et seq*., this Court has subject matter jurisdiction over the matters asserted herein under 28 U.S.C. §§ 1331 and 1338(a).

11. This Court has personal jurisdiction over Cisco, in part because Cisco does continuous and systematic business in this District, including by providing infringing products and services to the residents of the Western District of Texas that Cisco knew would be used within this District, and by soliciting business from the residents of the Western District of Texas. For example, Cisco is subject to personal jurisdiction in this Court because, *inter alia*, and on information and belief, Cisco has a regular and established place of business at its offices

in the Western District of Texas (and elsewhere in the State of Texas), and directly and through agents regularly does, solicits, and transacts business in the Western District of Texas (and elsewhere in the State of Texas), including, for example, through its www.cisco.com website.

12. In particular, Cisco has committed and continues to commit acts of infringement in violation of 35 U.S.C. § 271, and has made, used, marketed, distributed, offered for sale, sold, and/or imported infringing products in the State of Texas, including in this District, and engaged in infringing conduct within and directed at or from this District. For example, Cisco has purposefully and voluntarily placed the Accused Products into the stream of commerce with the expectation that the Accused Products will be used in this District. The Accused Products have been and continue to be distributed to and used in this District. Cisco's acts cause and have caused injury to Commstech, including within this District.

13. Venue is proper in this District under the provisions of 28 U.S.C. §§ 1391 and 1400(b) at least because a substantial part of the events or omissions giving rise to the claims occurred in this District, and because Cisco has committed acts of infringement in this District and has a regular and established place of business in this District.

PATENTS-IN-SUIT

The '340 Patent

14. U.S. Patent No. 6,349,340 ("the '340 Patent") is entitled "Data multicast channelization," and was issued on February 19, 2002. A true and correct copy of the '340 Patent is attached as Exhibit A.

15. The '340 Patent was filed on January 13, 2000 as U.S. Patent Application No. 09/482,496.

16. Commstech is the owner of all rights, title, and interest in and to the '340 Patent, with the

full and exclusive right to bring suit to enforce the '340 Patent, including the right to recover for past infringement.

17. The '340 Patent is valid and enforceable under United States Patent Laws.

18. The '340 Patent recognized several problems with existing high-speed network data distribution technology, such as multicast technology. Notably, the '340 Patent recognized that "[m]anagement of high-speed data across distributed data networks can involve two basic approaches," both of which have several drawbacks. Exhibit A at 1:32-33.

19. For instance, the '340 Patent recognized problems with a "more common approach" referred to as the "client-based" approach, where "client nodes notify server nodes of their interest in certain desired data," and the "servers can individually distribute data packets to each interested, subscribing client." Exhibit A at 1:33-39. In this respect, the '340 Patent recognized that this "client-based" approach "tends to overburden the server as network demands grow." *Id.* at 1:30-41. In particular, the '340 Patent discloses that "as additional client nodes are added to the network, the server not only must individually distribute the data packets to each interested client node, but also the server must individually distribute the data packets to each additional subscribing client node," and thus, "as the client node list grows, so does the server's workload." *Id.* at 1:41-47.

20. The '340 Patent also recognized problems with another approach referred to as the "server-based" approach that uses multicast technology, in which "the server transmits the data packet to a multicast destination address identifying a particular multicast session," and "[i]nterested client nodes merely subscribe to the multicast address, rather than the server, in order to receive the broadcast data." Exhibit A at 1:48-58. However, the '340 Patent recognized that "because all client nodes receive each broadcast data packet, regardless of

the content of the data packet, each client node must filter unwanted data upon receipt of each data packet," but "[c]lient nodes generally are uninterested in most of the broadcast data and, as a result, client nodes expend substantial processor resources identifying and discarding unwanted data packets." *Id.* at 1:54-2:4. Further, the '340 Patent recognized that, although these existing approaches "allow[ ] a server to provide data at high data transmission rates to more client[ ] nodes," these approaches can "limit the client node's ability to filter unwanted data packets" given the client node's "processor overhead." *Id.* at 2:7-11.

21. To address one or more shortcomings of existing high-speed network data distribution technology, such as existing multicast technology that "challeng[ed] the client node's ability to filter the unwanted data packets," the '340 Patent discloses, *inter alia*, a "method for efficient filtering of unwanted data in a multicast network environment" that "satisfies the long-felt need of the prior art by applying a combination hardware and software solution which selectively filters multicast data by selectively disabling channels containing unwanted data." Exhibit A at 2:14-25. The '340 Patent's "inventive arrangements" have "advantages over all other data distribution methods" and provide "a novel and nonobvious method for receiving the benefits of multicasting while avoiding the drawbacks associated with such systems." *Id.* at 2:26-30.

22. Indeed, the inventions of the '340 Patent improved the functionality of "client" computers operating in a multicast network environment by reducing the "substantial processor resources" expended by "client" computers using existing data filtering mechanisms, such as by reducing the resources expended by a "client" computer's "network applications software." Exhibit A at 6:9-47. In this respect, the inventions of the '340 Patent allow a

"client" computer to "avoid excessive software filtering" that leads to "performance gain" that can be "significant." *Id.* at 10:21-31.

The Inventions Claimed in the '340 Patent Improved Technology & Were Not Well-Understood, Routine, or Conventional

23. Given the state of the art at the time of the inventions of the '340 Patent, including the deficiencies in network data distribution systems of the time, the inventive concepts of the '340 Patent cannot be considered to be conventional, well-understood, or routine. *See, e.g.*, Exhibit A at 1:32-2:17. Indeed, there was a long-felt need in the art at the time of the inventions of the '340 Patent that the claimed inventions of the '340 Patent addressed. *See, e.g., id* at 2:20-26. In this respect, the '340 Patent discloses, among other things, an unconventional solution to problems arising in the context of network data distribution systems, namely, that "client" computers in such systems "expend[ed] substantial processor resources" filtering multicast data and this "processor overhead" inhibited the "client" computers' ability to handle the increasing user demands for network data distribution systems to broadcast more data. *See, e.g., id* at 2:1-17.

24. The inventions of the '340 Patent offered an unconventional, technological solution to such problems resulting in a "novel and nonobvious method for receiving the benefits of multicasting while avoiding the drawbacks associated with such [existing] systems." Exhibit A at 2:25-30; *see also, e.g., id.* at 10:21-26 ("The inventive multicast channelization strategy can increase the bandwidth available to the expanding client node base by distributing the broadcast data across multiple channels," such that "client nodes can selectively filter unwanted broadcast data within the network interface circuitry of each client node."). In this respect, the inventions of the '340 Patent improved the functionality

of "client" computers operating in a multicast network environment. *See, e.g.*, *id.* at 6:9-47, 10:21-31.

25. Indeed, it was not well-understood, routine, or conventional at the time of the inventions of the '340 Patent to perform the following functions, alone and/or in combination with one another: (i) selecting from among a plurality of multicast communications channels a source communications channel for receiving requested multicast data, (ii) enabling the selected source communications channel, (iii) receiving the requested multicast data through the enabled source communications channel, (iv) forwarding the requested multicast data to requesting processes, and (v) disabling the selected source communications channel when the requesting processes indicate that no further data is requested to be received over the selected source communications channel. *See, e.g.*, Exhibit A at Claims 1, 8, 14. Moreover, it was not well-understood, routine, or conventional at the time of the inventions of the '340 Patent to perform one or more of the following functions alone and/or in combination with one or more of the preceding functions: (i) receiving from one or more processes in a client node a request for multicast data, (ii) identifying a multicast data source for each requested data, and (iii) disabling an enabled selected source communications channel when the requesting client node process indicates that no further data is requested to be received from the identified multicast data source over the selected source communications channel and no other requesting client node processes have indicated a continuing need for further data to be received from the identified multicast data source over the selected source communications channel. *See, e.g.*, *id.* at Claims 1, 8, 14.

26. Further, it was not well-understood, routine, or conventional at the time of the inventions

of the '340 Patent to perform one or more of the following functions alone and/or in combination with one or more of the unconventional functions set forth in paragraph number 25: (i) filtering, from multicast data received through an enabled source communications channel, unwanted/unrequested multicast data, (ii) discarding the unwanted/unrequested multicast data, and (ii) forwarding the filtered multicast data to one or more requesting processes. *See, e.g.*, Exhibit A at Claims 3, 9, 15.

27. These are just exemplary reasons why the inventions claimed in the '340 Patent were not well-understood, routine, or conventional at the time of the invention of the '340 Patent.

28. Consistent with the problems addressed by the '340 Patent being rooted in network data distribution systems, the '340 Patent's inventions naturally are also rooted in that same technology that cannot be performed solely with pen and paper or in the human mind. Indeed, using pen and paper or a human mind would not only ignore, but would run counter to, the stated technical solution of the '340 Patent noted above and the technical problems that the '340 Patent was specifically designed to address. Likewise, at least because the '340 Patent's claimed inventions address problems rooted in network data distribution systems, these inventions are not merely drawn to longstanding human activities.

The '317 Patent

29. U.S. Patent No. 6,606,317 ("the '317 Patent") is entitled "Dual key controlled content addressable memory for accessing packet switch data buffer for multicasting data packets," and was issued on August 12, 2003. A true and correct copy of the '317 Patent is attached as Exhibit B.

30. The '317 Patent was filed on September 9, 1999 as U.S. Patent Application No. 09/391,919.

31. Commstech is the owner of all rights, title, and interest in and to the '317 Patent, with the full and exclusive right to bring suit to enforce the '317 Patent, including the right to recover for past infringement.

32. The '317 Patent is valid and enforceable under United States Patent Laws.

33. The '317 Patent recognized several problems with existing digital data management systems used to control the storage and distribution of high-speed digital data. Notably, the '317 Patent recognized that while "digital signal processing components have enabled telecommunication service providers to supply multiple types of signaling channels from one or more sourcing sites to a switching interface serving a number of destination equipment[]," the "need for increased storage and data delivery capacity of the data switching and distribution elements that make up the switching interface" have been inadequately addressed by existing systems that include "a very large (e.g., room-sized) data buffering subsystem, having separate (maximal capacity) data stores dedicated to each port being serviced." Exhibit B at 1:29-40. As a result of the "extraordinarily large size and considerable power requirements" of existing systems, the '317 Patent recognized that "this type of a data storage and distribution subsystem is not only impractical, but effectively impossible to deploy" in various environments, "where payload power consumption parameters must comply with very limited specifications." *Id.* at 1:40-46.

34. To address one or more shortcomings of existing digital data management systems, the '317 Patent discloses "a new and improved output centric packet switch architecture that employs a dual key content addressable memory (CAM)-based data storage management mechanism, which is configured to control, in a highly efficient manner, the storage and distribution of received data packets to one or more output ports of a P input port, M output

port packet switch." Exhibit B at 1:50:57. More specifically, the '317 Patent discloses that "[a] packet record is stored in only a single storage location of the output packet buffer," which "enables the capacity of the packet output buffer to be reduced considerably relative to the capacity of conventional data memories [that] store a separate copy of the data for each output port to which the packet is to be delivered." *Id.* at 2:11-17. In this regard, the '317 Patent discloses that this "storage space reduction" can be "particularly significant . . . ." *Id.* at 2:17-19. Moreover, the '317 Patent discloses that "[w]hen a packet is written to the output packet buffer, the header information is mapped . . . into a multi-field buffer address pointer word[] that is written to one or more (for multicasting) addresses of a dual key controlled content addressable memory (CAM)." *Id.* at 2:20-25. In this respect, "the amount of memory required to implement the CAM is considerably less than that of the output packet buffer." *Id.* at 2:36-38.

The Inventions Claimed in the '317 Patent Improved Technology & Were Not Well-Understood, Routine, or Conventional

35. Given the state of the art at the time of the inventions of the '317 Patent, including the deficiencies in digital data management systems of the time, the inventive concepts of the '317 Patent cannot be considered to be conventional, well-understood, or routine. *See, e.g.*, Exhibit B at 1:29-2:18. The '317 Patent discloses, among other things, an unconventional solution to problems arising in the context of digital data management systems, namely, that such systems provided an "impractical" or "effectively impossible" solution for certain environments where "payload power consumption parameters [had to] comply with very limited specifications." *See, e.g., id* at 1:40-46.

36. As noted above, the inventions of the '317 Patent offered an unconventional, technological solution to such problems resulting in a "a new and improved output centric packet switch

architecture that employs a dual key content addressable memory (CAM)-based data storage management mechanism, which is configured to control, in a highly efficient manner, the storage and distribution of received data packets to one or more output ports of a P input port, M output port packet switch." Exhibit B at 1:50:57. In this regard, the inventions of the '317 Patent improved the technical functioning of packet switches and computer communications systems. *See, e.g.*, *id.* at 1:50-3:23, 13:43-14:11. For instance, the inventions of the '317 Patent assign specific functions among a packet switch's components to, *inter alia*, improve the packet switch's memory storage and packet distribution capabilities. *See, e.g.*, *id*.

37. Indeed, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to store a data packet that is to be multicast to a plurality of recipients in only a single data packet storage location of a data packet buffer. *See, e.g.*, Exhibit B at Claims 1, 19, and 20. Moreover, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to store, in a content-addressable memory, a plurality of respectively different address pointer words, each address pointer word containing a respectively different key field that is used to identify one of a plurality of output ports, and an address field that identifies a single data packet storage location of a packet buffer in which the single data packet is stored. *See, e.g.*, *id.* at Claims 1, 10, 19, and 20. Further, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to couple a key to respectively different key fields of respectively different address pointer words stored in a content addressable memory, so as to access contents of the address field of an address pointer word whose key field matches the key. *See, e.g.*, *id.* at Claims 1, 10, and 19. Further yet, it was not well-understood,

routine, or conventional at the time of the inventions of the '317 Patent to perform a combination of one or more of the preceding functions. *See, e.g.*, *id.* at Claims 1, 19.

38. Furthermore, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to perform the following functions, alone and/or in combination with one another: (i) reading a data packet from a single data packet storage location of a packet buffer in accordance with accessed address field contents and (ii) coupling the accessed address field contents to address fields of respectively different address pointer words stored in a content-addressable memory to determine whether the single data packet storage location of the data packet buffer is available to store a new data packet. *See, e.g.*, Exhibit B at Claim 1. Moreover, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to perform a combination of one or more of the preceding functions with one or more of the unconventional functions set forth in paragraph number 37. *See, e.g.*, *id.* at Claims 1, 19.

39. It was also not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to perform the following functions, alone and/or in combination with one another: (i) coupling respectively different keys to respectively different first fields of respectively different address pointer words stored in a content addressable memory, to read out contents of a common address field for application to a packet buffer, and thereby cause a data packet to be read out from a single data packet storage location of the packet buffer and multicast to multiple ones of a plurality of output ports, and (ii) in the course of reading out the contents of the common address field for application to the packet buffer, coupling the contents of the common address field to address fields of all of the address pointer words stored in the content addressable memory, to determine whether the data

packet has been multicast in accordance with each address pointer word stored in the content addressable memory. *See, e.g.*, *id.* at Claim 19.

40. Additionally, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to have a content-addressable memory including multiple storage regions that contain one or more of the following components, alone and/or in combination with one another: (i) a first plurality of content addressable memory cells, the contents of which are associated with a first field of a respective address pointer word that identifies data to be accessed from a storage location in a memory, (ii) a second plurality of content addressable memory cells, the contents of which are associated with a second field of the respective address pointer word that identifies the address of the storage location in the memory, and (iii) wherein respectively different address pointer words stored in multiple ones of the plurality of storage regions of the content addressable memory contain respectively different first fields associated with respectively different ones of multiple instances of accessing the data from the storage location in the memory, and a common second field that identifies the address of the storage location in the memory for each of the multiple instances of accessing the data from the storage location in the memory. *See, e.g.*, Exhibit B at Claim 6.

41. Likewise, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to have a content-addressable memory that stores address information for controlling multiple accesses to the same data packet stored in a single memory address of a data packet output buffer, to selectively read out therefrom the same data packet to a plurality of ports of a high speed switch, the content addressable memory being configured to store respectively different buffer address pointer words that identify the same data

packet to be delivered to selected ones of a plurality of switch output ports, and point to the address of the single storage location of the packet buffer. *See, e.g.*, Exhibit B at Claim 20.

42. Further yet, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to have a content-addressable memory comprising an array of storage regions that store multibit words, each storage region being formed of a first plurality of content addressable memory cells, the contents of which are associated with a first field of a respective multibit word, and a second plurality of content addressable memory cells, the contents of which are associated with a second field of the respective multibit word, and wherein respectively different words stored in multiple ones of the plurality of storage regions of the content addressable memory contain respectively different first fields, and a common second field. *See, e.g.*, Exhibit B at Claim 22.

43. Moreover, it was not well-understood, routine, or conventional at the time of the inventions of the '317 Patent to have an output port data distribution architecture comprising one or more of the following components, alone and/or in combination with one another: (i) a packet buffer containing a plurality of storage locations that store data packets intended for delivery to one or more of a plurality of output ports, (ii) a content-addressable memory containing a plurality of storage regions that store respectively different address pointer words, each address pointer word containing a respectively different key field that is used to identify a data packet to be delivered to one of the plurality of output ports, and an address field that identifies the address of one of the plurality of storage locations of the packet buffer in which the data packet is stored, and (iii) a packet buffer access controller, which is operative to couple a key to key fields of address pointer words of the plural

storage regions of the content-addressable memory, and thereby access contents of the address field of an address pointer word whose key field contains said key, the accessed address field contents being coupled to read out a data packet stored in one of the plurality of storage locations of the packet buffer, the accessed address field contents being coupled to the address fields of address pointer words stored in the content-addressable memory, and wherein the content-addressable memory is operative to output a signal representative whether the accessed address field contents are contained in the address field of another address pointer word stored in the content-addressable memory. *See, e.g.*, Exhibit B at Claim 10.

44. These are just exemplary reasons why the inventions claimed in the '317 Patent were not well-understood, routine, or conventional at the time of the invention of the '317 Patent.

45. Consistent with the problems addressed being rooted in digital data management systems, the '317 Patent's inventions naturally are also rooted in that same technology that cannot be performed solely with pen and paper or in the human mind. Indeed, using pen and paper or a human mind would not only ignore, but would run counter to, the stated technical solution of the '317 Patent noted above and the technical problems that the '317 Patent was specifically designed to address. Likewise, at least because the '317 Patent's claimed inventions address problems rooted in digital data management systems, these inventions are not merely drawn to longstanding human activities.

The '946 Patent

46. U.S. Patent No. 7,126,946 ("the '946 Patent") is entitled "Dual key controlled content addressable memory for accessing packet switch data buffer for multicasting data packets," and was issued on October 24, 2006. A true and correct copy of the '946 Patent is attached

as Exhibit C.

47. The '946 Patent was filed on July 9, 2003 as U.S. Patent Application No. 10/616,286, which is a continuation of U.S. Patent Application No. 09/391,919, filed on September 9, 1999, and now the '317 Patent.

48. Commstech is the owner of all rights, title, and interest in and to the '946 Patent, with the full and exclusive right to bring suit to enforce the '946 Patent, including the right to recover for past infringement.

49. The '946 Patent is valid and enforceable under United States Patent Laws.

50. Commstech incorporates by reference and re-alleges the foregoing paragraph numbers 33-45 of this Complaint as if fully set forth herein.

51. Like the inventions claimed in the '317 Patent—the parent to the '946 Patent—the inventions claimed in the '946 Patent were not well-understood, routine, or conventional.

52. Indeed, it was not well-understood, routine, or conventional at the time of the invention of the '946 Patent to store data in a data memory by writing the data into a storage location of the data memory and writing, into a plurality of respective storage regions of a content-addressable memory, respective address pointer words, each of which includes a respective key field that is used to identify the data, and an address field that identifies the address of the storage location of the data memory. *See, e.g.,* Exhibit C at Claim 1. Moreover, it was not well-understood, routine, or conventional at the time of the invention of the '946 Patent to read data from a data memory by coupling a key to key fields of address pointer words stored in storage regions of the content-addressable memory, accessing an address of a storage location of the data memory from an address field of an address pointer word whose key field contains said key, and reading the data from the storage location of the data

memory in accordance with the accessed address. *See, e.g.*, *id.* at Claim 1. Moreover, it was not well-understood, routine, or conventional at the time of the invention of the '946 Patent to couple an accessed address to a content-addressable memory to determine whether the address of a storage location of data memory is contained in another address pointer word stored in the content-addressable memory. *See, e.g.*, *id.* at Claim 1. Further yet, it was not well-understood, routine, or conventional at the time of the inventions of the '946 Patent to perform a combination of one or more of the preceding functions. *See, e.g.*, *id.* at Claim 1.

53. These are just exemplary reasons why the inventions claimed in the '946 Patent were not well-understood, routine, or conventional at the time of the invention of the '946 Patent.

54. Consistent with the problems addressed being rooted in digital data management systems, the '946 Patent's inventions naturally are also rooted in that same technology that cannot be performed solely with pen and paper or in the human mind. Indeed, using pen and paper or a human mind would not only ignore, but would run counter to, the stated technical solution of the '946 Patent noted above and the technical problems that the '946 Patent was specifically designed to address. Likewise, at least because the '946 Patent's claimed inventions address problems rooted in digital data management systems, these inventions are not merely drawn to longstanding human activities.

The '231 Patent

55. U.S. Patent No. 7,152,231 ("the '231 Patent") is entitled "High Speed Interprocess Communication," and was issued on December 19, 2006. A true and correct copy of the '231 Patent is attached as Exhibit D.

56. The '231 Patent was filed on November 1, 1999 as U.S. Patent Application No. 09/431,449.

57. Commstech is the owner of all rights, title, and interest in and to the '231 Patent, with the full and exclusive right to bring suit to enforce the '231 Patent, including the right to recover for past infringement.

58. The '231 Patent is valid and enforceable under United States Patent Laws.

59. The '231 Patent recognized several problems with existing network interprocess communications ("IPC") technology. For instance, the '231 Patent recognized that, while existing IPC technology "provide[d] IPC between two processes," existing IPC technology was "not suitable for real time command and control systems which can require fail-safe and extremely fast conveyancing of information between process." Exhibit D at 2:35-39. In fact, the '231 Patent recognized that existing IPC technology "require[d] a minimum of two system calls to the operating system kernel," with each system call having the "risk[] [of] losing CPU control upon invoking the system call required to read or write [] data, respectively." *Id.* at 1:65-2:5. The '231 Patent further recognized that existing IPC technology "inefficiently pass[ed] data between processes by copying [] data stored in one region of memory, and storing the data in a different region of memory," which was "expensive in terms of processor overhead and time delay." *Id.* at 2:6-12, 2:39-44. The '231 Patent also recognized that existing IPC technology involving "pointer passing," where "a recipient process receives only an address of a location in memory of the message data," "become[s] unworkable because data residing at an address in one memory space is not equivalent to the data residing at the same address." *Id.* at 2:15-21, 2:25-30.

60. To address the shortcomings of existing IPC technology, which were not suitable for real-time command and control systems, the '231 Patent discloses an improved "system and method for high speed interprocess communications" that provides "extremely fast IPC

both by communicating message data in a shared region of random access memory (RAM) external to the operating system kernel and by limiting the movement of data." Exhibit D at 4:28-32. Notably, the '231 Patent discloses that "[p]rocesses are notified of the location of the message data rather than actually receiving a copy of the message data," and "[a]s a result, the number of data copies necessary for high speed IPC is minimized." *Id.* at 4:32-37. Moreover, the '231 Patent discloses that the inventions of the '231 Patent provided "significant differences" compared to existing IPC technology, such as (i) the inventions' "use of a shared region of RAM to store accumulated data" that resulted in (a) "not requir[ing] operating system calls to write and read accumulated data" and (b) "not requir[ing] the rebooting of the operating system" when the shared region of RAM is reconfigured, and (ii) the inventions providing "a faster and safer mechanism for IPC in that the overhead associated with the IPC is minimized from two system calls and 2n bytes of data movement to a minimal n bytes of data movement." *Id.* at 8:7-20.

The Inventions Claimed in the '231 Patent Improved Technology & Were Not Well-Understood, Routine, or Conventional

61. Given the state of the art at the time of the inventions of the '231 Patent, including the deficiencies in IPC systems of the time, the inventive concepts of the '231 Patent cannot be considered to be conventional, well-understood, or routine. *See, e.g.*, Exhibit D at 1:56-2:44, 4:10-46, 8:7-20. The '231 Patent discloses, among other things, an unconventional solution to problems arising in the context of IPC technology, namely, that such existing systems were not suitable for real-time command and control systems that required fast conveyancing of information between processes. *See, e.g.*, *id.* at 2:2-44

62. The inventions of the '231 Patent offered a technological solution to such problems resulting in a system that minimized the number of data copies necessary for high speed

IPC by providing mechanisms to communicate message data in a shared region of RAM that is external to the operating system kernel. Exhibit D at 4:28-37. In this respect, the inventions of the '231 Patent improved the technical functioning of one or more computers by reciting a specific technique for improving IPC. *See, e.g.*, *id.* at 4:11-37, 4:57-5:9.

63. Indeed, it was not well-understood, routine, or conventional at the time of the invention of the '231 Patent to have a system configured to attach a first and second process to a message buffer in a shared region of RAM that is exclusive of an operating system kernel space. *See, e.g.*, *id.* at Claims 1, 6, 10. Moreover, it was not well-understood, routine, or conventional at the time of the invention of the '231 Patent to have a system where a first process is configured to add a memory offset corresponding to a location in a message buffer to a message list of a second process, and manipulate message data in the second process at the location corresponding to the offset, where the message data is transferred from the first process to the second process with minimal data transfer overhead. *See, e.g.*, *id.* at Claims 1, 6.

64. Additionally, it was not well-understood, routine, or conventional at the time of the invention of the '231 Patent to have a system configured to (i) identify a memory offset in the message list corresponding to the second process, (ii) process, in the second process, message data stored at a location in the message buffer corresponding to the memory offset, and (iii) release the message buffer. *See, e.g.*, Exhibit D at Claims 4, 9, 13. It was also not well-understood, routine, or conventional at the time of the invention of the '231 Patent to have a system configured to lock the message data to prevent the first process from accessing the accumulated data while the message data is being manipulated. *See, e.g.*, *id.* at Claims 5, 14

65. Furthermore, it was not well-understood, routine, or conventional at the time of the invention of the '231 Patent to have a system configured to retrieve a memory offset in a message buffer corresponding to the location of message data accumulated by a first process and insert the memory offset in a message queue corresponding to a second process. *See, e.g.*, Exhibit D at Claims 2, 7, 11. It was also not well-understood, routine, or conventional at the time of the invention of the '231 Patent to have a system configured to atomically assign the memory offset to an integer location in the message queue corresponding to the second process. *See, e.g.*, *id.* at Claims 3, 8, 12.

66. These are just exemplary reasons why the inventions claimed in the '231 Patent were not well-understood, routine, or conventional at the time of the invention of the '231 Patent.

67. Consistent with the problems addressed being rooted in network IPC systems, the '231 Patent's inventions naturally are also rooted in that same technology that cannot be performed solely with pen and paper or in the human mind. Indeed, using pen and paper or a human mind would not only ignore, but would run counter to, the stated technical solution of the '231 Patent noted above and the technical problems that the '231 Patent was specifically designed to address. Likewise, at least because the '231 Patent's claimed inventions address problems rooted in network IPC systems, these inventions are not merely drawn to longstanding human activities.

The '028 Patent

68. U.S. Patent No. 7,769,028 ("the '028 Patent") is entitled "Systems and methods for adaptive throughput management for event-driven message-based data," and was issued on August 3, 2010. A true and correct copy of the '028 Patent is attached as Exhibit E.

69. The '028 Patent was filed on June 21, 2006 as U.S. Patent Application No. 11/471,923.

70. Commstech is the owner of all rights, title, and interest in and to the '028 Patent, with the full and exclusive right to bring suit to enforce the '028 Patent, including the right to recover for past infringement.

71. The '028 Patent is valid and enforceable under United States Patent Laws.

72. The '028 Patent discloses, among other things, "a method for communicating data including prioritizing data by assigning a priority to the data, analyzing a network to determine a status of the network, and communicating data based at least in part on the priority of the data and the status of the network." Exhibit E at Abstract. The '028 Patent also discloses "Quality of Service (QoS)," which "refers to one or more capabilities of a network to provide various forms of guarantees with regard to data this is carried." *Id.* at 4:16-18. The '028 Patent states that "[t]he primary goal of QoS is to provide priority including dedicated bandwidth, controlled jitter and latency (required by some real-time and interactive traffic), and improved [data] loss characteristics." *Id.* at 4:27-31.

73. In discussing QoS, the '028 Patent recognized various shortcomings of existing QoS systems. As one example, the '028 Patent states that "[e]xisting QoS systems cannot provide QoS based on message content at the transport layer" of the Open Systems Interconnection (OSI) seven-layer protocol model. Exhibit E at 5:1-2. Indeed, the '028 Patent explains that the "Transmission Control Protocol (TCP)," which is a protocol at the transport layer, "requires several forms of handshaking and acknowledgements to occur in order to send data," and "[h]igh latency and [data] loss may result in TCP hitting time outs and not being able to send much, if any, meaningful data over [] a network." *Id.* at 1:57-60, 3:53-57. As another example, the '028 Patent states that "[c]urrent approaches to QoS often require every node in a network to support QoS, or at the very least, for every node

in the network involved in a particular communication to support QoS," but such approaches to QoS "do[] not scale well because of the large amount of state information that must be maintained at every node and the overhead associated with setting up such connections." *Id.* at 4:35-39, 4:46-49. As yet another example, the '028 Patent states that "[d]ue to the mechanisms existing QoS solutions utilize, messages that look the same to current QoS systems may actually have different priorities based on message content," but "data consumers may require access to high-priority data without being flooded by lower-priority data." *Id.* at 4:61-67.

74. In discussing the shortcomings of the prior art, the '028 Patent recognized that "[t]here is a need for systems and methods for providing QoS on the edge of a [] data network," and "a need for adaptive, configurable QoS systems and methods in a [] data network." Exhibit E at 5:17-20. The claimed inventions of the '028 Patent provide such systems and methods.

The Inventions Claimed in the '028 Patent Improved Technology & Were Not Well-Understood, Routine, or Conventional

75. Given the state of the art at the time of the inventions of the '028 Patent, including the deficiencies with existing QoS systems for computer networks, the inventive concepts of the '028 Patent cannot be considered to be conventional, well-understood, or routine. *See, e.g.*, Exhibit E at 1:57-60, 3:53-57, 4:35-39, 4:46-49, 4:61-67, 5:1-2, 5:17-20. The '028 Patent discloses, among other things, an unconventional solution to problems arising in the context of communications networks that relied on existing QoS systems, namely, that such QoS systems did not scale, were not adaptive or configurable to different network types or architectures, and could not provide QoS based on message content at the transport layer, among other deficiencies. *See, e.g.*, *id.*

76. To address one or more deficiencies with existing QoS systems, the inventions of the '028

Patent offered a technological solution that facilitated providing an improved technique for communicating data over a network, which helped to control jitter and latency and improve data loss, among other benefits. In particular, the inventions of the '028 Patent provided a specific, unconventional solution for prioritizing data as part of and/or at the top of the transport layer, dynamically changing rules for assigning priority to data, and communicating data based at least in part on the priority of the data and the status of the network. *See, e.g.*, *id.* at Claims 1, 13, 17; 7:29-31. In this respect, the inventions of the '028 Patent improved the technical functioning of computers and computer networks by reciting a specific technique for prioritizing data communications over a network. *See, e.g.*, *id.* at 4:11-37, 4:57-5:9.

77. Indeed, it was not well-understood, routine, or conventional at the time of the invention of the '028 Patent for a communication device to (i) prioritize data by assigning priority to data, where the prioritization occurs either as part of and/or at the top of the transport layer, (ii) analyze a network to determine a status of the network, (iii) select a mode based on the status of the network, (iv) change rules for assigning priority to the data based on the mode, and (v) communicate the data based at least in part on the priority of the data and the status of the network, where the data is communicated at a transmission rate metered based at least in part on the status of the network. *See, e.g.*, Exhibit E at Claim 1. Moreover, it was not well-understood, routine, or conventional at the time of the invention of the '028 Patent for a communication device to receive the data at a node on the edge of the network. *See, e.g.*, Exhibit E at Claim 5. It was also not well-understood, routine, or conventional at the time of the invention of the '028 Patent for a communication device to receive the data at least in part from an application program and/or communicate the data to an application

program. *See, e.g.*, *id.* at Claims 6, 12. Further, it was not well-understood, routine, or conventional at the time of the invention of the '028 Patent for a communication device to assign the priority to the data based at least in part on message content of the data, protocol information of the data, or a user defined rule. *See, e.g.*, *id.* at Claims 7-9.

78. Additionally, it was not well-understood, routine, or conventional at the time of the invention of the '028 Patent for a communication system to include (i) a data prioritize component adapted to assign a priority to data, where the prioritization occurs either as part of and/or at the top of the transport layer, (ii) a network analysis component adapted to determine a status of the network, (iii) a mode selection component adapted to select a mode based at least on the status of the network, and (iv) a data communications component adapted to communicate the data based at least in part on the priority of the data and the status of the network, where the data prioritization component is adapted to assign priority to the data based on prioritization rules that are selected based on a selected mode, and where the data is communicated at a transmission rate metered based at least in part on the status of the network. *See, e.g.*, Exhibit E at Claims 13, 17. It was also not well-understood, routine, or conventional at the time of the invention of the '028 Patent for a communication system to include a data organization component adapted to organize the data with respect to other data based at least in part on the priority of the data. *See, e.g.*, *id.* at Claim 14.

79. These are just exemplary reasons why the inventions claimed in the '028 Patent were not well-understood, routine, or conventional at the time of the invention of the '028 Patent.

80. Consistent with the problems addressed being rooted in QoS systems for computer networks, the '028 Patent's inventions naturally are also rooted in that same technology that cannot be performed solely with pen and paper or in the human mind. Indeed, using

pen and paper or a human mind would not only ignore, but would run counter to, the stated technical solution of the '028 Patent noted above and the technical problems that the '028 Patent was specifically designed to address. Likewise, at least because the '028 Patent's claimed inventions address problems rooted in QoS systems for computer networks, these inventions are not merely drawn to longstanding human activities.

The '860 Patent

81. U.S. Patent No. 7,990,860 ("the '860 Patent") is entitled "Method and system for rule-based sequencing for QoS," and was issued on August 2, 2011. A true and correct copy of the '860 Patent is attached as Exhibit F.

82. The '860 Patent was filed on June 16, 2006 as U.S. Patent Application No. 11/454,220.

83. Commstech is the owner of all rights, title, and interest in and to the '860 Patent, with the full and exclusive right to bring suit to enforce the '860 Patent, including the right to recover for past infringement.

84. The '860 Patent is valid and enforceable under United States Patent Laws.

85. The '860 Patent discloses, among other things, "a method for communicating data over a network to provide Quality of Service," including "prioritizing the data, and communicating the data based at least in part on the priority." Exhibit F at Abstract. According to the '860 Patent, "Quality of Service (QoS)" "refers to one or more capabilities of a network to provide various forms of guarantees with regard to data that is carried." *Id.* at 4:16-18. The '860 Patent states that "[t]he primary goal of QoS is to provide priority including dedicated bandwidth, controlled jitter and latency (required by some real-time and interactive traffic), and improved [data] loss characteristics." *Id.* at 4:27-32.

86. Like the '028 Patent, the '860 Patent recognized various shortcomings of existing QoS

systems. As one example, the '860 Patent states that "[e]xisting QoS systems cannot provide QoS based on message content at the transport layer" of the Open Systems Interconnection (OSI) seven-layer protocol model. Exhibit F at 5:2-3. Indeed, the '860 Patent explains that the "Transmission Control Protocol (TCP)," which is a protocol at the transport layer, "requires several forms of handshaking and acknowledgements to occur in order to send data," and "[h]igh latency and [data] loss may result in TCP hitting time outs and not being able to send much, if any, meaningful data over [] a network." *Id.* at 1:57-60, 3:53-57. As another example, the '860 Patent states that "[c]urrent approaches to QoS often require every node in a network to support QoS, or at the very least, for every node in the network involved in a particular communication to support QoS," but such approaches to QoS "do[] not scale well because of the large amount of state information that must be maintained at every node and the overhead associated with setting up such connections." *Id.* at 4:36-39, 4:47-50. As yet another example, the '860 Patent states that "[d]ue to the mechanisms existing QoS solutions utilize, messages that look the same to current QoS systems may actually have different priorities based on message content," but "data consumers may require access to high-priority data without being flooded by lower-priority data." *Id.* at 4:64-5:1.

87. In discussing the shortcomings of the prior art, the '860 Patent recognized that "[t]here is a need for systems and methods for providing QoS on the edge of a [] data network," and "a need for adaptive, configurable QoS systems and methods in a [] data network." Exhibit F at 5:19-22. The claimed inventions of the '860 Patent provide such systems and methods.

The Inventions Claimed in the '860 Patent Improved Technology & Were Not Well-Understood, Routine, or Conventional

88. Given the state of the art at the time of the inventions of the '860 Patent, including the

deficiencies with existing QoS systems for computer networks, the inventive concepts of the '860 Patent cannot be considered to be conventional, well-understood, or routine. *See, e.g.*, Exhibit F at 1:57-60, 3:53-57, 4:36-39, 4:47-50, 4:64-5:2, 5:19-22. The '860 Patent discloses, among other things, an unconventional solution to problems arising in the context of communications networks that relied on existing QoS systems, namely, that such QoS systems did not scale, were not adaptive or configurable to different network types or architectures, and could not provide QoS based on message content at the transport layer, among other deficiencies. *See, e.g.*, *id.*

89. To address one or more deficiencies with existing QoS systems, the inventions of the '860 Patent offered a technological solution that facilitated providing an improved technique for communicating data over a network, which helped to control jitter and latency and improve data loss, among other benefits. In particular, the inventions of the '860 Patent provided a specific, unconventional solution for prioritizing data as part of and/or at the top of the transport layer by sequencing the data based at least in part on a user defined rule. *See, e.g.*, *id.* at Abstract, Claims 1, 13, 17. In this respect, the inventions of the '860 Patent improved the technical functioning of computers and computer networks by reciting a specific technique for prioritizing data communications over a network. *See, e.g.*, *id.* at 4:11-37, 4:57-5:9.

90. Indeed, it was not well-understood, routine, or conventional at the time of the invention of the '860 Patent for a communication device to include (i) a network analysis component configured to determine a network status from a plurality of network statuses based on analysis of network measurements, and determine at least one of an effective link speed and a link proportion for at least one link, (ii) a mode selection component configured to

select a mode from a plurality of modes that corresponds with at least one of the plurality of network statuses based on the determined network status, where each of the plurality of modes comprises a user defined sequencing rule, (iii) a data prioritization component configured to operate at a transport layer of a protocol stack and prioritize the data by assigning a priority to the data, where the prioritization component includes a sequencing component configured to sequence the data based at least in part on the user defined sequencing rule of the selected mode, (iv) a data metering component configured to meter inbound data by shaping the inbound data at the data communications system for the at least one link, and meter outbound data by policing the outbound data at the data communications system for the at least one link, and (v) a data communication component configured to communicate the data based at least in part on the priority of the data, the effective link speed, and/or the link proportion. *See, e.g.*, Exhibit F at Claims 1, 15, 20.

91. Moreover, it was not well-understood, routine, or conventional at the time of the invention of the '860 Patent for the user defined sequencing rule mentioned above to be dynamically reconfigurable. *See, e.g.*, Exhibit F at Claim 5. It was also not well-understood, routine, or conventional at the time of the invention of the '860 Patent for a communication device to receive the data at least in part from an application program operating on the node, or pass the data at least in part to an application program operating on the node. *See, e.g.*, *id.* at Claims 6, 12. Further, it was not well-understood, routine, or conventional at the time of the invention of the '860 Patent for a communication device to prioritize the data by differentiating the data based at least in part on message content, protocol information, or a user defined differentiation rule. *See, e.g.*, *id.* at Claims 8-11.

92. These are just exemplary reasons why the inventions claimed in the '860 Patent were not

well-understood, routine, or conventional at the time of the invention of the '860 Patent.

93. Consistent with the problems addressed being rooted in QoS systems for computer networks, the '860 Patent's inventions naturally are also rooted in that same technology that cannot be performed solely with pen and paper or in the human mind. Indeed, using pen and paper or a human mind would not only ignore the stated technical solution of the '860 Patent noted above and the technical problem that the '860 Patent was specifically designed to address. Likewise, at least because the '860 Patent's claimed inventions address problems rooted in QoS systems for computer networks, these inventions are not merely drawn to longstanding human activities.

COUNT I: INFRINGEMENT OF U.S. PATENT NO. 6,349,340

94. Commstech incorporates by reference and re-alleges paragraphs 14-28 of this Complaint as if fully set forth herein.

95. Defendant Cisco has infringed and is infringing, either literally or under the doctrine of equivalents, the '340 Patent in violation of 35 U.S.C. § 271 *et seq.*, directly and/or indirectly, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license, products that support the RFC 4607 specification related to "Source-Specific Multicast for IP" (e.g., Cisco devices that operate with the Cisco NX-OS software, such as Cisco Nexus series switches, Cisco MDS 9000 Family storage switches, and Cisco UCS 6100 Series Fabric Interconnects) (collectively referred to herein as the "Accused '340 Products"). *See, e.g.,* https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/nx-os-software/data_sheet_c78-652063.html?dtid=osscdc000283.

96. As just one non-limiting example, set forth below (with claim language in bold and italics)

is exemplary evidence of infringement of Claim 1 of the '340 Patent in connection with the Accused '340 Products. This description is based on publicly available information. Commstech reserves the right to modify this description, including, for example, on the basis of information about the Accused '340 Products that it obtains during discovery.

1(a): A method for receiving requested multicast data over a plurality of multicast communications channels comprising:—Cisco makes, uses, sells, and/or offers to sell a device or system that practices the method of receiving requested multicast data over a plurality of multicast communications channels in accordance with Claim 1.

For instance, Cisco devices that operate with the Cisco NX-OS software (e.g., Cisco Nexus series switches, Cisco MDS 9000 Family storage switches, and Cisco UCS 6100 Series Fabric Interconnects) support the RFC 4607 specification related to "Source-Specific Multicast for IP" that discloses the method recited in Claim 1. *See, e.g.,* https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/nx-os-software/data_sheet_c78-652063.html?dtid=osscdc000283 (expressly disclosing "RFC 4607" as one or many "supported standards" for "Cisco NX-OS"); Holbrook, Source-specific multicast for IP, RFC 4607 (2006), pp. 3-5, *available at* https://tools.ietf.org/pdf/rfc4607.pdf. In particular, RFC 4607 defines a "source-specific multicast service" ("SSM") as "[a] datagram sent with source IP address S and destination IP address G in the SSM range [that] is delivered to each host socket that has specifically requested delivery of datagrams sent by S to G, and only to those sockets." Holbrook, Source-specific multicast for IP, RFC 4607 (2006), p.5, *available at* https://tools.ietf.org/pdf/rfc4607.pdf.

1(b): selecting from among the plurality of multicast communications channels a source

communications channel for receiving said requested multicast data;—Cisco makes, uses, sells, and/or offers to sell a device or system that selects from among the plurality of multicast communications channels a source communications channel for receiving said requested multicast data.

For instance, Cisco devices that operate with the Cisco NX-OS software (e.g., Cisco Nexus series switches, Cisco MDS 9000 Family storage switches, and Cisco UCS 6100 Series Fabric Interconnects) support the RFC 4607 specification, which discloses a plurality of multicast communication channels, where each "channel is identified (addressed) by the combination of a unicast source address and a multicast destination address in the SSM range" (e.g., "S, G = (192.0.2.1, 232.7.8.9)," "S, G = (192.0.2.2, 232.7.8.9)"). *Id.* at p. 6; *see also, e.g., id.* at pp. 3-4 ("The network service identified by (S,G), for SSM address G and source host address S, is referred to as a 'channel'"); *id.* at p. 6 ("We use the term 'channel' to refer to the service associated with an SSM address," and "[a] channel is identified by the combination of an SSM destination address and a specific source, e.g., an (S,G) pair."). In particular RFC 4607 discloses that "[t]he IP module interface to upper-layer protocols is extended to allow a socket to 'Subscribe' to . . . a particular channel identified by an SSM destination address and a source IP address." *Id.* at p. 5; *see also, e.g., id.* at p. 6 ("The receiver operations allowed on a channel are called 'Subscribe (S,G)' and 'Unsubscribe (S,G)'"); *id.* at p. 7 ("If reception of the same channel is desired on multiple interfaces, Subscribe is invoked once for each"); *id.* at p. 8 ("An incoming datagram destined to an SSM address MUST be delivered by the IP module to all sockets that have indicated (via Subscribe) a desire to receive data that matches the datagram's source address, destination address, and arriving interface.").

1(c): enabling said selected source communications channel;—Cisco makes, uses, sells, and/or offers to sell a device or system that enables the selected source communications channel.

For instance, Cisco devices that operate with the Cisco NX-OS software (e.g., Cisco Nexus series switches, Cisco MDS 9000 Family storage switches, and Cisco UCS 6100 Series Fabric Interconnects) support the RFC 4607 specification, which discloses that "[t]he IP module interface to upper-layer protocols is extended to allow a socket to 'Subscribe' to . . . a particular channel identified by an SSM destination address and a source IP address," and subscribing to a particular channel comprises selecting a source communications channel and also enabling the selected source communications channel. *Id.* at p. 5; *see also, e.g., id.* at p. 6 ("The receiver operations allowed on a channel are called 'Subscribe (S,G)' and 'Unsubscribe (S,G)'"); *id.* at p. 7 ("If reception of the same channel is desired on multiple interfaces, Subscribe is invoked once for each"); *id.* at p. 8 ("An incoming datagram destined to an SSM address MUST be delivered by the IP module to all sockets that have indicated (via Subscribe) a desire to receive data that matches the datagram's source address, destination address, and arriving interface."). Indeed, RFC 4607 discloses that "'interface' is a local identifier of the network interface on which reception of the channel identified by the (source-address, group-address) pair is to be ***enabled*** [e.g., subscribed] or disabled [e.g., unsubscribed]." *Id.* at p. 7 (emphasis added).

1(d): receiving said requested multicast data through said enabled source communications channel;—Cisco makes, uses, sells, and/or offers to sell a device or system that receives the requested multicast data through the enabled source communications channel.

For instance, Cisco devices that operate with the Cisco NX-OS software (e.g., Cisco Nexus series switches, Cisco MDS 9000 Family storage switches, and Cisco UCS 6100 Series Fabric Interconnects) support the RFC 4607 specification, which discloses that "[a]n incoming datagram destined to an SSM address MUST be delivered by the IP module to all sockets that have indicated (via Subscribe) a desire to receive data that matches the datagram's source address, destination address, and arriving interface." *Id.* at p. 8; *see also, e.g., id.* ("When the first socket on host H subscribes to a channel (S,G) on interface I, the host IP module on H sends a request on interface I to indicate to neighboring routers that the host wishes to receive traffic sent by source S to source-specific multicast destination G.").

1(e): forwarding said requested multicast data to requesting processes; and,—Cisco makes, uses, sells, and/or offers to sell a device or system that forwards the requested multicast data to requesting processes.

For instance, as noted above, Cisco devices that operate with the Cisco NX-OS software (e.g., Cisco Nexus series switches, Cisco MDS 9000 Family storage switches, and Cisco UCS 6100 Series Fabric Interconnects) support the RFC 4607 specification, which discloses that "[a]n incoming datagram destined to an SSM address MUST be delivered by the IP module to all ***sockets*** that have indicated (via Subscribe) a desire to receive data that matches the datagram's source address, destination address, and arriving interface." *Id.* at p. 8 (emphasis); *see also, e.g., id.* ("When the first socket on host H subscribes to a channel (S,G) on interface I, the host IP module on H sends a request on interface I to indicate to neighboring routers that the host wishes to receive traffic sent by source S to source-specific multicast destination G."). In particular, RFC 4607 defines a "socket" as "an

implementation-specific parameter used to distinguish among different requesting entities (e.g., programs or ***processes*** or communication end-points within a program or process) within the requesting host." *Id.* at p. 5.

1(f): disabling said selected source communications channel when said requesting processes indicate that no further data is requested to be received over said selected source communications channel.—Cisco makes, uses, sells, and/or offers to sell a device or system that disables the selected source communications channel when the requesting processes indicate that no further data is requested to be received over the selected source communications channel.

For instance, Cisco devices that operate with the Cisco NX-OS software (e.g., Cisco Nexus series switches, Cisco MDS 9000 Family storage switches, and Cisco UCS 6100 Series Fabric Interconnects) support the RFC 4607 specification, which discloses that "[t]he IP module interface to upper-layer protocols is extended to allow a socket to . . . 'Unsubscribe' from a particular channel identified by an SSM destination address and a source IP address," and unsubscribing from a particular channel disables the particular channel to indicate that no further data is requested to be received over the particular channel. *Id.* at p. 5; *see also, e.g., id.* at p. 8 (disclosing that "[a]n incoming datagram destined to an SSM address MUST be delivered by the IP module to all sockets that have indicated (via Subscribe) a desire to receive data that matches the datagram's source address, destination address, and arriving interface," but "MUST NOT be delivered to other sockets" (e.g., sockets that have Unsubscribed)). Indeed, as noted above, RFC 4607 discloses that "'interface' is a local identifier of the network interface on which reception of the channel identified by the (source-address, group-address) pair is to be enabled [e.g.,

subscribed] or ***disabled*** [e.g., unsubscribed]." *Id.* at p. 7 (emphasis added).

97. Additionally, Defendant Cisco has been and/or currently is an active inducer of infringement of the '340 Patent under 35 U.S.C. § 271(b) and contributory infringer of the '340 Patent under 35 U.S.C. § 271(c).

98. Cisco knew of the '340 Patent, or at least should have known of the '340 Patent, but was willfully blind to its existence. On information and belief, Cisco has had actual knowledge of the '340 Patent since at least as early as the filing and/or service of this Complaint.

99. Cisco has provided the Accused '340 Products to its customers and, on information and belief, instructions to use the Accused '340 Products in an infringing manner while being on notice of (or willfully blind to) the '340 Patent and Cisco's infringement. Therefore, on information and belief, Cisco knew or should have known of the '340 Patent and of its own infringing acts, or deliberately took steps to avoid learning of those facts.

100. Cisco knowingly and intentionally encourages and aids at least its end-user customers to directly infringe the '340 Patent.

101. Cisco's end-user customers directly infringe at least one or more claims of the '340 Patent by using the Accused '340 Products in their intended manner to infringe. Cisco induces such infringement by providing the Accused '340 Products and instructions to enable and facilitate infringement, knowing of, or being willfully blind to the existence of, the '340 Patent. On information and belief, Cisco specifically intends that its actions will result in infringement of one or more claims of the '340 Patent, or subjectively believe that their actions will result in infringement of the '340 Patent, but took deliberate actions to avoid learning of those facts, as set forth above.

102. Additionally, Cisco contributorily infringes at least one or more claims of the '340 Patent

by providing the Accused '340 Products and/or software components thereof, that embody a material part of the claimed inventions of the '340 Patent, that are known by Cisco to be specially made or adapted for use in an infringing manner, and are not staple articles with substantial non-infringing uses. The Accused '340 Products are specially designed to infringe at least one or more claims of the '340 Patent, and their accused components have no substantial non-infringing uses. In particular, on information and belief, the software modules and code that implement and perform the infringing functionalities identified above are specially made and adapted to carry out said functionality and do not have any substantial non-infringing uses.

103. At least as early as the filing and/or service of this Complaint, Cisco's infringement of the '340 Patent was and continues to be willful and deliberate, entitling Commstech to enhanced damages.

104. Additional allegations regarding Cisco's knowledge of the '340 Patent and willful infringement will likely have evidentiary support after a reasonable opportunity for discovery.

105. Cisco's infringement of the '340 Patent is exceptional and entitles Commstech to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

106. Commstech is in compliance with any applicable marking and/or notice provisions of 35 U.S.C. § 287 with respect to the '340 Patent.

107. Commstech is entitled to recover from Cisco all damages that Commstech has sustained as a result of Cisco's infringement of the '340 Patent, including, without limitation, a reasonable royalty.

COUNT II: INFRINGEMENT OF U.S. PATENT NO. 6,606,317

108. Commstech incorporates by reference and re-alleges paragraphs 29-45 of this Complaint as if fully set forth herein.

109. Defendant Cisco has infringed and is infringing, either literally or under the doctrine of equivalents, the '317 Patent in violation of 35 U.S.C. § 271 *et seq.*, directly and/or indirectly, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license, products that operate with the Cisco IOS software and comprise Content Addressable Memory (CAM) and/or Ternary Content Addressable Memory (TCAM) (including but not limited to Cisco Catalyst series switches) (collectively referred to herein as the "Accused '317 Products"), that infringe at least one or more claims of the '317 Patent.

110. As just one non-limiting example, set forth below is exemplary evidence of infringement of Claim 6 of the '317 Patent in connection with the Accused '317 Products. This description is based on publicly available information. Commstech reserves the right to modify this description, including, for example, on the basis of information about the Accused '317 Products that it obtains during discovery.

6(a): A content-addressable memory having a plurality of storage regions that store respectively different address pointer words, a respective storage region containing:— Cisco makes, uses, sells, and/or offers to sell a device that comprises a CAM and/or a TCAM having a plurality of storage regions that store respectively different address pointer words, where a respective storage region contains the elements recited in Claim 6.

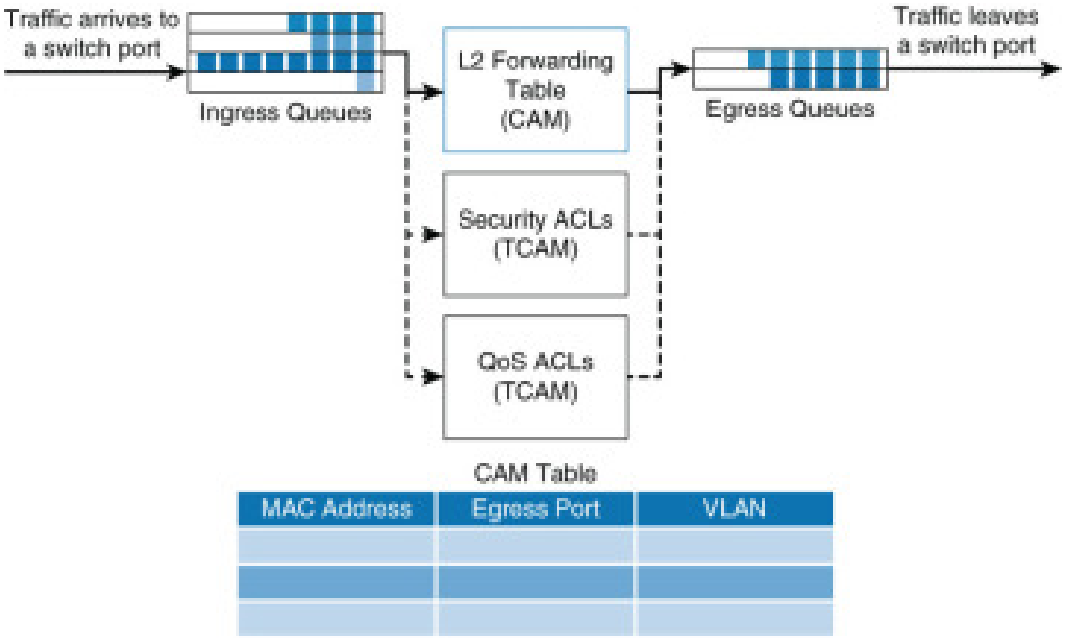
For instance, "Cisco Catalyst switches deploys [ ] memory tables using specialized memory architectures, referred to as CAM and TCAM." *See* https://community.cisc

o.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938. Specifically, Cisco discloses that "[f]or the MAC table, switches use content-addressable memory (CAM), whereas the ACL and QoS tables are housed in ternary content-addressable memory (TCAM)," both of which include a plurality of respective storage regions to write respective address pointer words. *See* http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2; *see also, e.g.,* https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white-paper-c11-737405.html; https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938 ("Catalyst switch architecture supports the ability to perform multiple lookups into multiple distinct CAM and TCAM regions in parallel."); https://www.pagiamtzis.com/cam/camintro/ ("The match address output of the CAM is in fact a pointer used to retrieve associated data from the RAM. . . . The CAM/RAM search can be viewed as a dictionary lookup where the search data is the word to be queried and the RAM contains the word definitions.").

6(b): a first plurality of content addressable memory cells, the contents of which are associated with a first field of a respective address pointer word that identifies data to be accessed from a storage location in a memory; and—Cisco makes, uses, sells, and/or offers to sell a device that comprises a CAM and/or a TCAM having a plurality of storage regions that store respectively different address pointer words, where a respective storage region comprises a first plurality of content addressable memory cells, the contents of which are associated with a first field of a respective address pointer word that identifies data to be accessed from a storage location in a memory.

For instance, "Cisco Catalyst switches deploys [ ] memory tables using specialized memory architectures, referred to as CAM and TCAM." *See* https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938.

With regard to its CAM architecture, one example of a memory table used by Cisco in the Accused '317 Products is set forth below:

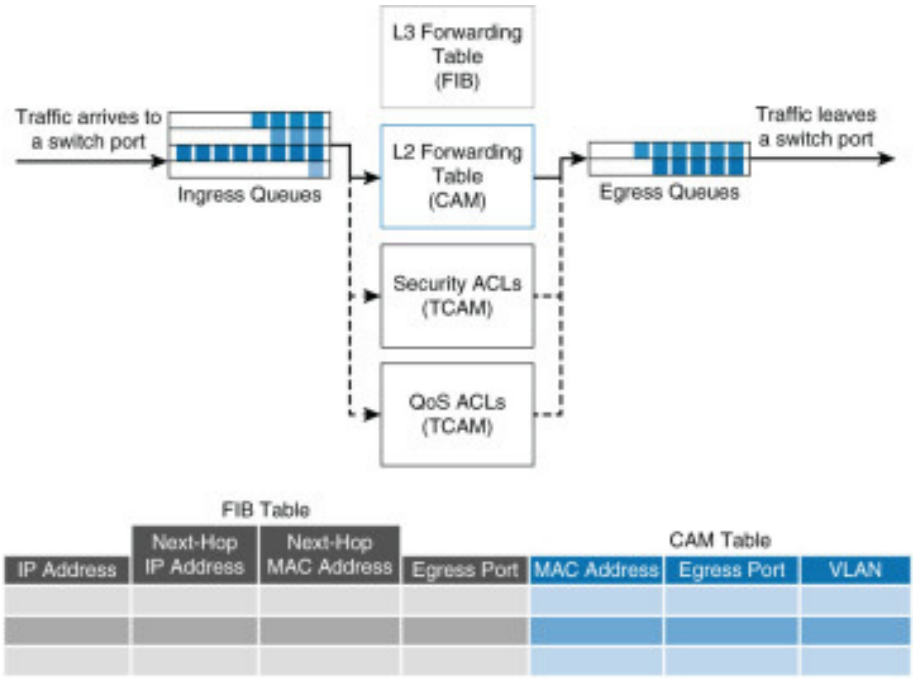


http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2.

Cisco explains that this type of table "also called the MAC table, contains information about where to forward the frame. Specifically, it contains MAC addresses and destination ports. The switches reference the destination MAC address of the incoming frame in the MAC table and forward the frames to the destination ports specified in the table. If the MAC address is not found, the frame is flooded through all ports in the same VLAN." *Id.; see also, e.g.,* https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938 ("The information a

switch uses to perform a lookup in a CAM table is called a key. For example, a Layer 2 lookup would use a destination MAC address and a VLAN ID as a key.")

With regard to its TCAM architecture, one example of a memory table used by Cisco in the Accused '317 Products is set forth below:

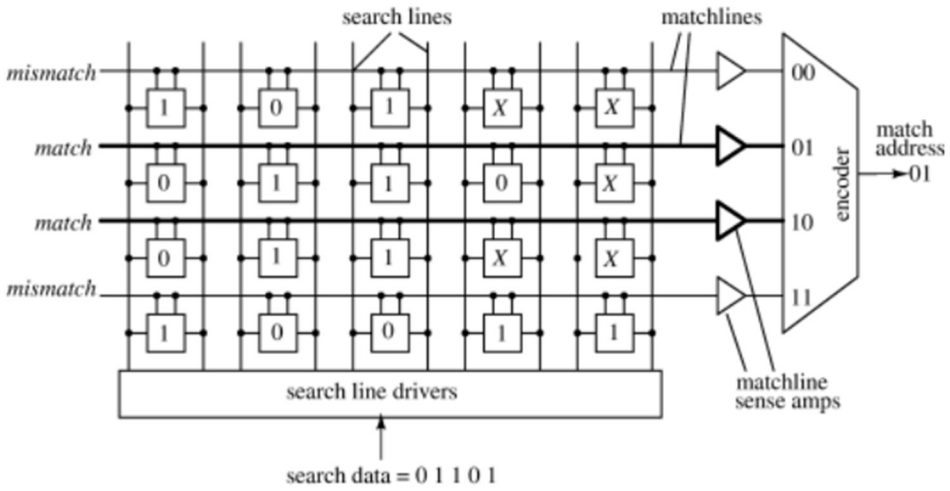


http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2.

Cisco explains that "[e]ach classification TCAM entry is 144 bits wide, and uses a lookup key to initiate a lookup into the TCAM." https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white-paper-c11-737405.html; *see also, e.g.,* https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938 ("The information a switch uses to perform a lookup in a CAM table is called a key. For example, a Layer 2 lookup would use a destination MAC address and a VLAN ID as a key."). Cisco further explains that, "[t]his lookup key uses input fields such as the ACL label (which is obtained from the LIF table), packet type (IPv4, IPv6, and more) and other fields to generate the key," and

"[t]he result of the TCAM lookup provides a pointer into the classification SRAM that holds the actual ACE entry." *Id.; see also, e.g.,* http://www.ciscopress.com/articles/article.asp?p=101629&seqNum=4 ("TCAM also uses a table lookup operation . . . . For example, binary values (0s and 1s) make up a key into the table, but a mask value is also used to decide which bits of the key are actually relevant. . . . This effectively makes a key consisting of three input values: 0, 1, and X (don't care) bit values—a three-fold or *ternary* combination."); https://www.pagiamtzis.com/cam/camintro/ ("The match address output of the CAM is in fact a pointer used to retrieve associated data from the RAM. . . . The CAM/RAM search can be viewed as a dictionary lookup where the search data is the word to be queried and the RAM contains the word definitions.").

One general example of a CAM of the type used by Cisco in the Accused '317 Products is shown in the figure below and comprises "core cells [that] are arranged into four horizontal words, each five bits longs," "*search lines* [that] run vertically in the figure and broadcast the *search data* to the CAM cells," "*matchlines* [that] run horizontally across the array and indicate whether the search data matches the row's word:



See https://www.pagiamtzis.com/cam/camintro/; *see also, e.g.,* http://www.ciscopress.c

om/articles/article.asp?p=2348265&seqNum=2 *(citing* https://www.pagiamtzis.com/cam/camintro/).

6(c): a second plurality of content addressable memory cells, the contents of which are associated with a second field of said respective address pointer word that identifies the address of said storage location in said memory, and—Cisco makes, uses, sells, and/or offers to sell a device that comprises a CAM and/or a TCAM having a plurality of storage regions that store respectively different address pointer words, where a respective storage region comprises a second plurality of content addressable memory cells, the contents of which are associated with a second field of said respective address pointer word that identifies the address of said storage location in said memory. *See, e.g.,* citations for claim element 6(b) above.

Indeed, Cisco explains that, "[t]he Layer 3 forwarding engine provides for a dual lookup into each bank, allowing for four lookups to be performed simultaneously. This means that for each input packet, up to four classification rules can be matched during IFE (ingress) processing, and up to four classification rules can be matched during OFE (egress) processing." https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white-paper-c11-737405.html. And in particular, Cisco further explains that, "[w]hen performing a Layer 3 lookup, the FIB TCAM lookup is performed first. To execute the lookup, a FIB TCAM lookup key is derived, based on incoming packet type and other fields, to perform a FIB TCAM lookup. The result of the FIB lookup returns a pointer into the FIB RLDRAM, which will hold a pointer into the adjacency table for normal forwarded packets. If the adjacency pointer indicates the destination can be reached through multiple paths, it computes a unique adjacency pointer for each path." *Id.*

6(d):wherein respectively different address pointer words stored in multiple ones of said plurality of storage regions of said content addressable memory contain respectively different first fields associated with respectively different ones of multiple instances of accessing said data from said storage location in said memory, and a common second field that identifies the address of said storage location in said memory for each of said multiple instances of accessing said data from said storage location in said memory.— In line with the discussion above, Cisco makes, uses, sells, and/or offers to sell a device that comprises a CAM and/or a TCAM in which respectively different address pointer words stored in multiple ones of the plurality of storage regions of the content addressable memory contain respectively different first fields associated with respectively different ones of multiple instances of accessing the data from the storage location in the memory, and a common second field that identifies the address of the storage location in the memory for each of the multiple instances of accessing the data from the storage location in the memory. *See, e.g.,* citations for claim elements 6(b) and 6(c) above.

Additionally, on information and belief, the Accused '317 Products comprise a CAM and/or a TCAM in which respectively different address pointer words are or can be stored in multiple ones of the plurality of storage regions of the content addressable memory contain respectively different first fields associated with respectively different ones of multiple instances of accessing the data from the storage location in the memory, and a common second field that identifies the address of the storage location in the memory for each of the multiple instances of accessing the data from the storage location in the memory as a result of having the particular architecture described above and also as a result of being configured to engage in the specific functionality described above.

111. Additionally, Defendant Cisco has been and/or currently is an active inducer of infringement of the '317 Patent under 35 U.S.C. § 271(b) and contributory infringer of the '317 Patent under 35 U.S.C. § 271(c).

112. Cisco knew of the '317 Patent, or at least should have known of the '317 Patent, but was willfully blind to its existence. On information and belief, Cisco has had actual knowledge of the '317 Patent since at least as early as the filing and/or service of this Complaint.

113. Cisco has provided the Accused '317 Products to its customers and, on information and belief, instructions to use the Accused '317 Products in an infringing manner while being on notice of (or willfully blind to) the '317 Patent and Cisco's infringement. Therefore, on information and belief, Cisco knew or should have known of the '317 Patent and of its own infringing acts, or deliberately took steps to avoid learning of those facts.

114. Cisco knowingly and intentionally encourages and aids at least its end-user customers to directly infringe the '317 Patent.

115. Cisco's end-user customers directly infringe at least one or more claims of the '317 Patent by using the Accused '317 Products in their intended manner to infringe. Cisco induces such infringement by providing the Accused '317 Products and instructions to enable and facilitate infringement, knowing of, or being willfully blind to the existence of, the '317 Patent. On information and belief, Cisco specifically intends that its actions will result in infringement of at least one or more claims of the '317 Patent, or subjectively believe that their actions will result in infringement of the '317 Patent, but took deliberate actions to avoid learning of those facts, as set forth above.

116. Additionally, Cisco contributorily infringes at least one or more claims of the '317 Patent by providing the Accused '317 Products and/or software components thereof, that embody

a material part of the claimed inventions of the '317 Patent, that are known by Cisco to be specially made or adapted for use in an infringing manner, and are not staple articles with substantial non-infringing uses. The Accused '317 Products are specially designed to infringe at least one or more claims of the '317 Patent, and their accused components have no substantial non-infringing uses. In particular, on information and belief, the software modules and code that implement and perform the infringing functionalities identified above are specially made and adapted to carry out said functionality and do not have any substantial non-infringing uses.

117. At least as early as the filing and/or service of this Complaint, Cisco's infringement of the '317 Patent was and continues to be willful and deliberate, entitling Commstech to enhanced damages.

118. Additional allegations regarding Cisco's knowledge of the '317 Patent and willful infringement will likely have evidentiary support after a reasonable opportunity for discovery.

119. Cisco's infringement of the '317 Patent is exceptional and entitles Commstech to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

120. Commstech is in compliance with any applicable marking and/or notice provisions of 35 U.S.C. § 287 with respect to the '317 Patent.

121. Commstech is entitled to recover from Cisco all damages that Commstech has sustained as a result of Cisco's infringement of the '317 Patent, including, without limitation, a reasonable royalty.

COUNT III: INFRINGEMENT OF U.S. PATENT NO. 7,126,946

122. Commstech incorporates by reference and re-alleges paragraphs 46-55 of this Complaint

as if fully set forth herein.

123. Defendant Cisco has infringed and is infringing, either literally or under the doctrine of equivalents, the '946 Patent in violation of 35 U.S.C. § 271 *et seq.*, directly and/or indirectly, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license, products that operate with the Cisco IOS software and comprise Content Addressable Memory (CAM) and/or Ternary Content Addressable Memory (TCAM) (including but not limited to Cisco Catalyst series switches that comprise a Switching Supervisor Engine) (collectively referred to herein as the "Accused '946 Products"), that infringe at least one or more claims of the '946 Patent.

124. As just one non-limiting example, set forth below is exemplary evidence of infringement of Claim 1 of the '946 Patent in connection with the Accused '946 Products. This description is based on publicly available information. Commstech reserves the right to modify this description, including, for example, on the basis of information about the Accused '946 Products that it obtains during discovery.

1(a): A method of interfacing data with a data memory comprising the steps of:—Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS software to perform the method of interfacing data with a data memory comprising the steps recited in Claim 1.

For instance, Cisco discloses that "Catalyst switches maintain several types of tables to be used in the switching process," and "[t]he tables are tailored for Layer 2 switching or MLS [multilayer switching], and are kept in very fast memory so that many fields within a frame or packet can be compared in parallel." *See* http://www.ciscopress.com/articles/article.asp?p=101629&seqNum=4. In particular,

"[r]outing, switching, ACL [Access Control List] and QoS tables are stored in a high-speed table memory so that forwarding decisions and restrictions can be made in high-speed hardware," and "[s]witches perform lookups in these tables for result information, such as to determine whether a packet with a specific destination IP address is supposed to be dropped according to an ACL." *See* https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938. Notably, Cisco discloses that "[a]ll Catalyst switch models use a Content Addressable Memory (CAM) table for Layer 2 switching," where "the source MAC addresses are learned and recorded in the CAM table" as frames arrive on switch ports, and these Catalyst switch models also comprise a "Ternary Content Addressable Memory (TCAM)" which is an "extension of the CAM table concept." *See* http://www.ciscopress.com/articles/article.asp?p=101629&seqNum=4.

1(b): storing said data in said data memory by: writing said data into a storage location of said data memory, and —Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS software to perform the function of storing data in data memory by writing the data into a storage location of the data memory.

For instance, Cisco Catalyst switches that operate with the Cisco IOS software comprise Random-Access Memory ("RAM") that is configured to read and write data into a storage location. *See, e.g.,* https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938 ("In the case of ordinary RAM the IOS uses a memory address to get the data stored at this memory location, while with CAM the IOS does the inverse. It uses the data and the CAM returns the address where the data is stored."); https://www.cisco.com/c/en/us/products/

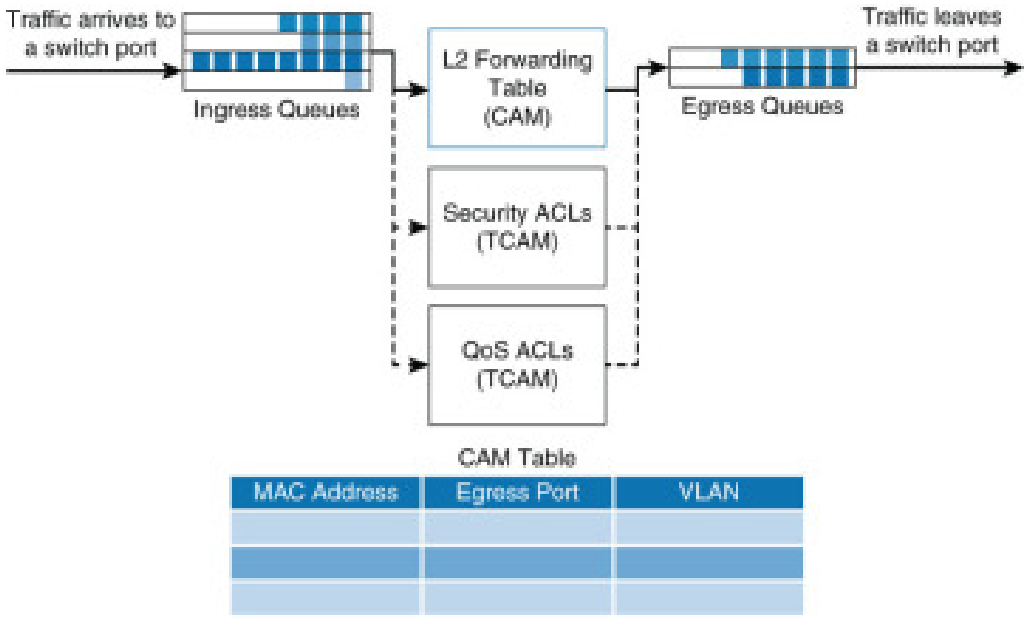
collateral/switches/catalyst-6500-series-switches/white-paper-c11-737405.html ("Integrates external SRAMs . . . Uses 4 sets of 32K x 96-bit eDRAM . . . Full ECC with an additional 8 bits on read/write), ("The result of the TCAM lookup provides a pointer into the classification SRAM that holds the actual ACE entry."); https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white_paper_c11-676346.html (disclosing a "Layer 3 Forwarding Engine" that comprises a "Classification SRAM"); https://www.pagiamtzis.com/cam/camintro/ ("The match address output of the CAM is in fact a pointer used to retrieve associated data from the RAM."); http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2 (citing https://www.pagiamtzis.com/cam/camintro/).

1(c): writing, into a plurality of respective storage regions of a content-addressable memory, respective address pointer words, each of which includes a respective key field that is used to identify said data, and an address field that identifies the address of said storage location of said data memory;—Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS software to perform the function of storing data in data memory by writing, into a plurality of respective storage regions of a content-addressable memory, respective address pointer words, each of which includes a respective key field that is used to identify said data, and an address field that identifies the address of said storage location of said data memory.

For instance, Cisco Catalyst switches that operate with the Cisco IOS software "use specialized hardware to house the MAC table, ACL lookup data, and QoS lookup data." *See* http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2. More specifically, Cisco discloses that "[f]or the MAC table, switches use content-addressable

memory (CAM), whereas the ACL and QoS tables are housed in ternary content-addressable memory (TCAM)," both of which include a plurality of respective storage regions to write respective address pointer words. *See id.*; *see also, e.g.,* https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white-paper-c11-737405.html; https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938 ("Catalyst switch architecture supports the ability to perform multiple lookups into multiple distinct CAM and TCAM regions in parallel.").

With regard to its CAM architecture, one example of a memory table used by Cisco in the Accused '946 Products is set forth below:

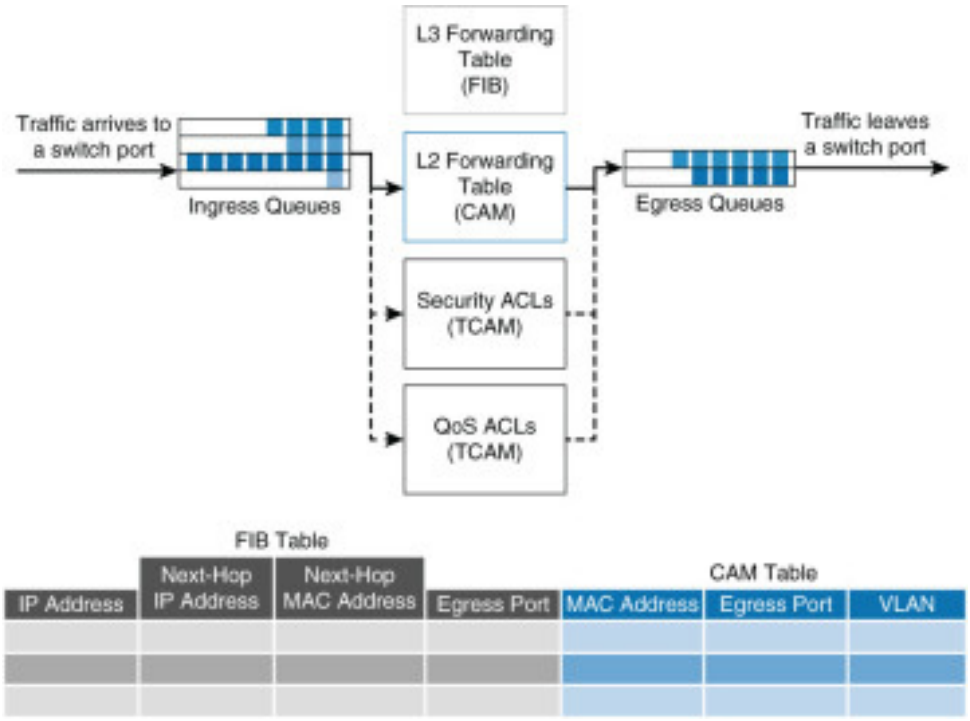


http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2.

Cisco explains that this type of table "also called the MAC table, contains information about where to forward the frame. Specifically, it contains MAC addresses and destination ports. The switches reference the destination MAC address of the incoming frame in the MAC table and forward the frames to the destination ports specified in the

table. If the MAC address is not found, the frame is flooded through all ports in the same VLAN." *Id.; see also, e.g.,* https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938 ("The information a switch uses to perform a lookup in a CAM table is called a key. For example, a Layer 2 lookup would use a destination MAC address and a VLAN ID as a key.")

With regard to its TCAM architecture, one example of a memory table used by Cisco in the Accused '946 Products is set forth below:



http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2. Cisco explains that "[e]ach classification TCAM entry is 144 bits wide, and uses a lookup key to initiate a lookup into the TCAM." https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white-paper-c11-737405.html. In particular, "[t]his lookup key uses input fields such as the ACL label (which is obtained from the LIF table), packet type (IPv4, IPv6, and more) and other fields to generate the key," and "[t]he

result of the TCAM lookup provides a pointer into the classification SRAM that holds the actual ACE entry." *Id; see also, e.g.,* http://www.ciscopress.com/articles/article.asp?p=101629&seqNum=4 ("TCAM also uses a table lookup operation . . . . For example, binary values (0s and 1s) make up a key into the table, but a mask value is also used to decide which bits of the key are actually relevant. . . . This effectively makes a key consisting of three input values: 0, 1, and X (don't care) bit values—a three-fold or *ternary* combination."); https://www.pagiamtzis.com/cam/camintro/ ("The match address output of the CAM is in fact a pointer used to retrieve associated data from the RAM. . . . The CAM/RAM search can be viewed as a dictionary lookup where the search data is the word to be queried and the RAM contains the word definitions.").

1(d): reading said data from said data memory by: (b1) coupling a key to key fields of address pointer words stored in storage regions of said content-addressable memory, and accessing said address of said storage location of said data memory from the address field of an address pointer word whose key field contains said key;—Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS software to perform the function of reading the data from the data memory by coupling a key to key fields of address pointer words stored in storage regions of the content-addressable memory, and accessing the address of the storage location of the data memory from the address field of an address pointer word whose key field contains the key.

For instance, as noted above, for Cisco Catalyst switches that operate with the Cisco IOS software, Cisco discloses that "[e]ach classification TCAM entry is 144 bits wide, and uses a lookup key to initiate a lookup into the TCAM." https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-

switches/white-paper-c11-737405.html. In particular, "[t]his lookup key uses input fields such as the ACL label (which is obtained from the LIF table), packet type (IPv4, IPv6, and more) and other fields to generate the key," and "[t]he result of the TCAM lookup provides a pointer into the classification SRAM that holds the actual ACE entry." *See* http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2. As Cisco explains:

> When the ACL engine needs to perform a lookup on a packet, it creates a lookup key. The lookup key contains the same type of information as the patterns in the TCAM (IP addresses, ports, and so on) but is based on the contents of the packet passing through the switch.
>
> The ACL engine scans the patterns in the TCAM in parallel to see if the lookup key matches. The key matches if the appropriate bits in the key match all the bits in the pattern that are masked with a match bit (using the mask associated with that pattern). Bits that are masked with a don't care bit are ignored—bits in the lookup key do not have to match those bits. The result returned is always the longest match in the TCAM.

See https://www.ndm.net/ips/pdf/cisco/Catalyst-6500/65acl_wp.pdf; *see also, e.g.*, https://www.pagiamtzis.com/cam/camintro/ ("The match address output of the CAM is in fact a pointer used to retrieve associated data from the RAM. . . . The CAM/RAM search can be viewed as a dictionary lookup where the search data is the word to be queried and the RAM contains the word definitions.").

Moreover, Cisco explains that, "[t]he Layer 3 forwarding engine provides for a dual lookup into each bank, allowing for four lookups to be performed simultaneously. This means that for each input packet, up to four classification rules can be matched during IFE (ingress) processing, and up to four classification rules can be matched during OFE (egress) processing." https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white-paper-c11-737405.html. And in particular, Cisco

further explains that, "[w]hen performing a Layer 3 lookup, the FIB TCAM lookup is performed first. To execute the lookup, a FIB TCAM lookup key is derived, based on incoming packet type and other fields, to perform a FIB TCAM lookup. The result of the FIB lookup returns a pointer into the FIB RLDRAM, which will hold a pointer into the adjacency table for normal forwarded packets. If the adjacency pointer indicates the destination can be reached through multiple paths, it computes a unique adjacency pointer for each path." *Id.*

1(e): (b2) reading said data from said storage location of said data memory in accordance with said address accessed in step (b1); and—Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS software to perform the function of reading the data from the storage location of the data memory in accordance with the address accessed in element 1(d) of Claim 1.

For instance, as noted above, for Cisco Catalyst switches that operate with the Cisco IOS software, Cisco discloses that "[t]he result of the TCAM lookup provides a pointer into the classification SRAM that holds the actual ACE entry," and the Cisco Catalyst switches can read the data from the storage location of the SRAM in accordance with the address accessed in element 1(d) of Claim 1. *See* http://www.ciscopress.com/articles/article.asp?p=2348265&seqNum=2.; *see also, e.g.,* https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938 ("In the case of ordinary RAM the IOS uses a memory address to get the data stored at this memory location, while with CAM the IOS does the inverse. It uses the data and the CAM returns the address where the data is stored."); https://www.pagiamtzis.com/cam/camintro/ ("The match address output of the CAM is in fact a pointer used

to retrieve associated data from the RAM. In this case the associated data is the output port.").

1(f): coupling said address accessed in step (b1) to said content-addressable memory, to determine whether said address of said storage location of said data memory is contained in another address pointer word stored in said content-addressable memory.—Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS software to perform the function of coupling the address accessed in element 1(d) of Claim 1 to the content-addressable memory, to determine whether the address of the storage location of the data memory is contained in another address pointer word stored in the content-addressable memory.

For instance, as noted above, for Cisco Catalyst switches that operate with the Cisco IOS software, Cisco explains that "[t]he ACL engine scans the patterns in the TCAM in parallel to see if the lookup key matches," where [t]he key matches if the appropriate bits in the key match all the bits in the pattern that are masked with a match bit (using the mask associated with that pattern)." *See* https://www.ndm.net/ips/pdf/cisco/Catalyst-6500/65acl_wp.pdf; *see also, e.g.,* https://community.cisco.com/t5/networking-documents/cam-content-addressable-memory-vs-tcam-ternary-content/ta-p/3107938 ("Catalyst switch architecture supports the ability to perform multiple lookups into multiple distinct CAM and TCAM regions in parallel."), ("The information a switch uses to perform a lookup in a CAM table is called a key. For example, a Layer 2 lookup would use a destination MAC address and a VLAN ID as a key.").

Moreover, the Cisco Catalyst switches that operate with the Cisco IOS software are configured to utilize a "mls acl tcam share-global" command that "enables [a] static sharing

feature. With static sharing, only one copy of the PACL/ACL and inherited VLAN-based feature ACLs is stored in the TCAM for all ports using the same ACL set, freeing TCAM space for more ACLs." https://community.cisco.com/t5/networking-documents/understanding-mls-acl-tcam-share-global-command/ta-p/3117551. As such, the Cisco Catalyst switches that operate with the Cisco IOS software couple the address accessed in element 1(d) of Claim 1 to the content-addressable memory, to determine whether the address of the storage location of the data memory is contained in another address pointer word stored in the content-addressable memory

Additionally, on information and belief, the Cisco Catalyst switches that operate with the Cisco IOS software perform the function of coupling the address accessed in element 1(d) of Claim 1 to the content-addressable memory, to determine whether the address of the storage location of the data memory is contained in another address pointer word stored in the content-addressable memory as a result of having the particular architecture described above and also as a result of being configured to engage in the specific functionality described above.

125. Additionally, Defendant Cisco has been and/or currently is an active inducer of infringement of the '946 Patent under 35 U.S.C. § 271(b) and contributory infringer of the '946 Patent under 35 U.S.C. § 271(c).

126. Cisco knew of the '946 Patent, or at least should have known of the '946 Patent, but was willfully blind to its existence. On information and belief, Cisco has had actual knowledge of the '946 Patent since at least as early as the filing and/or service of this Complaint.

127. Cisco has provided the Accused '946 Products to its customers and, on information and belief, instructions to use the Accused '946 Products in an infringing manner while being

on notice of (or willfully blind to) the '946 Patent and Cisco's infringement. Therefore, on information and belief, Cisco knew or should have known of the '946 Patent and of its own infringing acts, or deliberately took steps to avoid learning of those facts.

128. Cisco knowingly and intentionally encourages and aids at least its end-user customers to directly infringe the '946 Patent.

129. Cisco's end-user customers directly infringe at least one or more claims of the '946 Patent by using the Accused '946 Products in their intended manner to infringe. Cisco induces such infringement by providing the Accused '946 Products and instructions to enable and facilitate infringement, knowing of, or being willfully blind to the existence of, the '946 Patent. On information and belief, Cisco specifically intends that its actions will result in infringement of at least one or more claims of the '946 Patent, or subjectively believe that their actions will result in infringement of the '946 Patent, but took deliberate actions to avoid learning of those facts, as set forth above.

130. Additionally, Cisco contributorily infringes at least one or more claims of the '946 Patent by providing the Accused '946 Products and/or software components thereof, that embody a material part of the claimed inventions of the '946 Patent, that are known by Cisco to be specially made or adapted for use in an infringing manner, and are not staple articles with substantial non-infringing uses. The Accused '946 Products are specially designed to infringe at least one or more claims of the '946 Patent, and their accused components have no substantial non-infringing uses. In particular, on information and belief, the software modules and code that implement and perform the infringing functionalities identified above are specially made and adapted to carry out said functionality and do not have any substantial non-infringing uses.

131. At least as early as the filing and/or service of this Complaint, Cisco's infringement of the '946 Patent was and continues to be willful and deliberate, entitling Commstech to enhanced damages.

132. Additional allegations regarding Cisco's knowledge of the '946 Patent and willful infringement will likely have evidentiary support after a reasonable opportunity for discovery.

133. Cisco's infringement of the '946 Patent is exceptional and entitles Commstech to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

134. Commstech is in compliance with any applicable marking and/or notice provisions of 35 U.S.C. § 287 with respect to the '946 Patent.

135. Commstech is entitled to recover from Cisco all damages that Commstech has sustained as a result of Cisco's infringement of the '946 Patent, including, without limitation, a reasonable royalty.

COUNT IV: INFRINGEMENT OF U.S. PATENT NO. 7,152,231

136. Commstech incorporates by reference and re-alleges paragraphs 55-67 of this Complaint as if fully set forth herein.

137. On information and belief, Defendant Cisco has infringed and is infringing, either literally or under the doctrine of equivalents, the '231 Patent in violation of 35 U.S.C. § 271 *et seq.*, directly and/or indirectly, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license, products that operate with the Cisco IOS XR software (including but not limited to Cisco CRS series routers, XR 12000 series routers, and ASR 9000 series routers) (collectively referred to herein as the "Accused '231 Products"), that infringe at least one or more claims of the

'231 Patent. *See, e.g.*, https://www.cisco.com/en/US/docs/ios_xr_sw/Beta2_R3.3/plf_33_B.pdf; Mobeen Tahir et al., *Cisco IOS XR Fundamentals* (Cisco Press, 2009), pp. 13-14, *available at* http://the-eye.eu/public/Books/IT%20Various/cisco_ios_xr_fundamentals.pdf.

138. As just one non-limiting example, set forth below is exemplary evidence of infringement of Claim 1 of the '231 Patent in connection with the Accused '231 Products. This description is based on publicly available information. Commstech reserves the right to modify this description, including, for example, on the basis of information about the Accused '231 Products that it obtains during discovery.

1(a): A method for high speed interprocess communications comprising the steps of:—On information and belief, Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS XR software to perform a method for high speed interprocess communications comprising in accordance with Claim 1.

For example, the Cisco IOS XR software implements interprocess communication ("IPC") through a method known generally in the art as "shared memory." Cisco's implementation of IPC shared memory in the Cisco IOS XR software constitutes a method for high speed interprocess communications. More specifically, the Cisco IOS XR software provides a "microkernel-based operating system" that comprises "functionalities including memory management, task scheduling, synchronization services, context switching, and interprocess communication (IPC)." Mobeen Tahir et al., *Cisco IOS XR Fundamentals* (Cisco Press, 2009), pp. 9-10.

1(b): detecting a previously created shared region of RAM;—Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS XR software to perform the

function of detecting a previously created shared region of RAM.

For instance, Cisco generally discloses that "some forms of communication between processes are better handled using *shared memory,* which is accessible by multiple processes," and "operating system[s] provide[] different synchronization mechanisms between processes that are writing to or reading from shared memory regions." *Id.* at p. 8; *see also, e.g.,* https://community.cisco.com/t5/service-providers-documents/ncs6000-shared-memory-faq/ta-p/3160317 ("In computer hardware, shared memory refers to a (typically large) block of random access memory (RAM) that can be accessed by several different central processing units (CPUs) in a multiple-processor computer system."). In particular, Cisco discloses that the IOS XR software "monitors processes to detect CPU hog and memory depletion," and also "monitors disk space utilization, deadlocks, kernel thread, file descriptors, and shared memory usage." Mobeen Tahir et al., *Cisco IOS XR Fundamentals* (Cisco Press, 2009), p. 53. Cisco further discloses that "access to the shared memory space must be synchronized to ensure data consistency," and that the Cisco IOS XR software provides "mutex, condvar, and semaphore synchronization tools" to ensure such data consistency. *Id.* at pp. 23-24; *see also id.* at p. 350 (disclosing a "Shared memory timestamp").

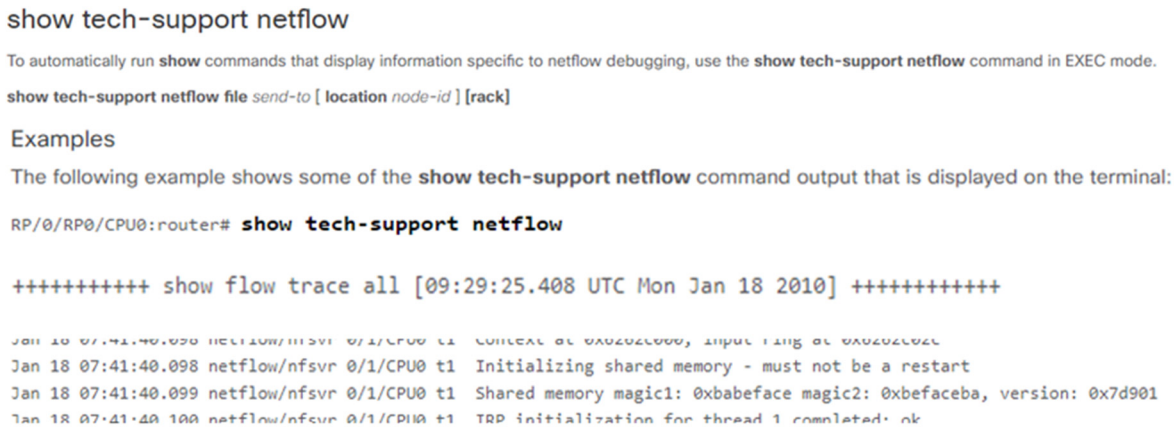
Moreover, the Cisco IOS XR software provides various commands to manage processes and memory, including a "show memory" command to "display the available physical memory and memory usage information of processes on [a] router," including the "shared memory." *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/routers/crs/software/crs_r5-3/sysman/command/reference/b-sysman-cr53xcrs/b-sysman-cr53xcrs_chapter_01101.html.

1(c): if a shared region of RAM is not detected, creating and configuring a shared region of RAM for storing accumulated data;—On information and belief, Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS XR software to perform the functions of creating and configuring a shared region of RAM for storing accumulated data if a shared region of RAM is not detected.

For instance, Cisco generally discloses that "some forms of communication between processes are better handled using *shared memory,* which is accessible by multiple processes," and "operating system[s] provide[] different synchronization mechanisms between processes that are writing to or reading from shared memory regions." Mobeen Tahir et al., *Cisco IOS XR Fundamentals* (Cisco Press, 2009), p. 8; *see also, e.g.,* https://community.cisco.com/t5/service-providers-documents/ncs6000-shared-memory-faq/ta-p/3160317 ("In computer hardware, shared memory refers to a (typically large) block of random access memory (RAM) that can be accessed by several different central processing units (CPUs) in a multiple-processor computer system."). In particular, Cisco discloses that the IOS XR software "monitors processes to detect CPU hog and memory depletion," and also "monitors disk space utilization, deadlocks, kernel thread, file descriptors, and shared memory usage." Mobeen Tahir et al., *Cisco IOS XR Fundamentals* (Cisco Press, 2009), p. 53. Cisco further discloses that "access to the shared memory space must be synchronized to ensure data consistency," and that Cisco IOS XR software provides "mutex, condvar, and semaphore synchronization tools" to ensure such data consistency. *Id.* at pp. 23-24; *see also id.* at 350 (disclosing a "Shared memory timestamp").

Moreover, as shown below, the Cisco IOS XR software provides various

commands to manage processes and memory, including a "show tech-support netflow" command to "display information specific to netflow debugging" that includes information indicating that "shared memory" is initialized:



show tech-support netflow

To automatically run **show** commands that display information specific to netflow debugging, use the **show tech-support netflow** command in EXEC mode.

show tech-support netflow file *send-to* [**location** *node-id*] **[rack]**

Examples

The following example shows some of the **show tech-support netflow** command output that is displayed on the terminal:

```
RP/0/RP0/CPU0:router# show tech-support netflow

++++++++++++ show flow trace all [09:29:25.408 UTC Mon Jan 18 2010] ++++++++++++

Jan 18 07:41:40.098 netflow/nfsvr 0/1/CPU0 t1  Initializing shared memory - must not be a restart
Jan 18 07:41:40.099 netflow/nfsvr 0/1/CPU0 t1  Shared memory magic1: 0xbabeface magic2: 0xbefaceba, version: 0x7d901
```

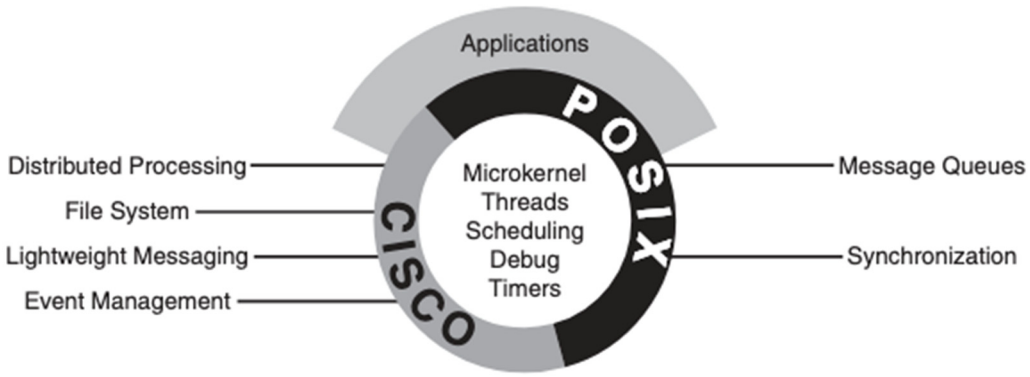
See, e.g., Cisco IOS XR Advanced System Command Reference for the Cisco CRS Router, pp. 451-454, *available at* https://dokumen.tips/documents/cisco-ios-xr-advanced-system-command-reference-for-the-cisco-crs-router-release.html.

1(d): attaching first and second processes to a message buffer in the shared region of random access memory (RAM) exclusive of operating system kernel space, each said process having a message list that is a message queue;—On information and belief, Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS XR software to perform the function of attaching first and second processes to a message buffer in the shared region of RAM exclusive of operating system kernel space, where each process has a message list that is a message queue.

For instance, on information and belief, the Cisco IOS XR provides the capability to attach processes to a message buffer in the shared memory that resides outside the operating system kernel space. Indeed, Cisco generally discloses that "[i]n a microkernel system," such as the Cisco IOS XR operating system, "only essential core OS services

reside inside the kernel," and "[a]ll other services, including device drivers and network drivers, reside in their own address space." Mobeen Tahir et al., *Cisco IOS XR Fundamentals* (Cisco Press, 2009), p. 8. In particular, Cisco discloses that "[t]he microkernel used in Cisco IOS XR . . . is lightweight and does not include system services such as device drivers, file systems, and network stack." *Id.* at p. 10; *see also, e.g., id.* at p. 15 ("Because each process outside the microkernel is restartable without impacting the rest of the system, failure of a process due to memory corruption of software defect does not impact other parts of the system.").

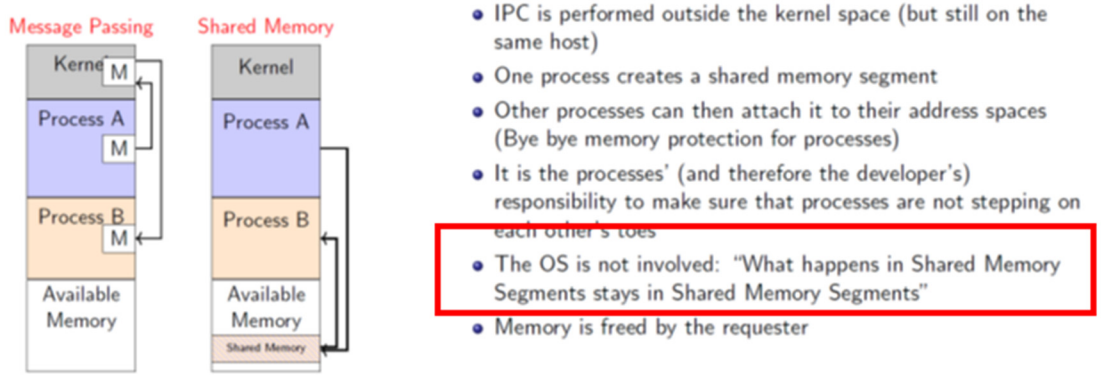
Moreover, Cisco discloses that each process has a message list (e.g., a "linked list") that is a message queue. *See, e.g., id.* at pp. 23-24. In particular, Cisco discloses that "access to shared memory space must be synchronized to ensure data consistency," and in one specific example, Cisco explains that "if one thread attempts to access a linked list while another thread is in the process of updating it, the result could be catastrophic." *Id.* Cisco further discloses a visual representation of the IOS XR software that includes "Message Queues":



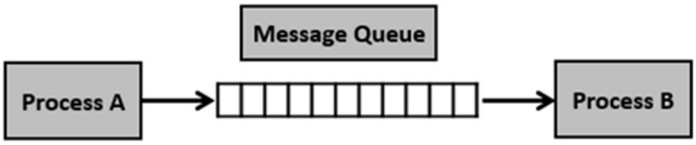
See id. at p. 160.

Moreover, attaching first and second processes to a message buffer in the shared

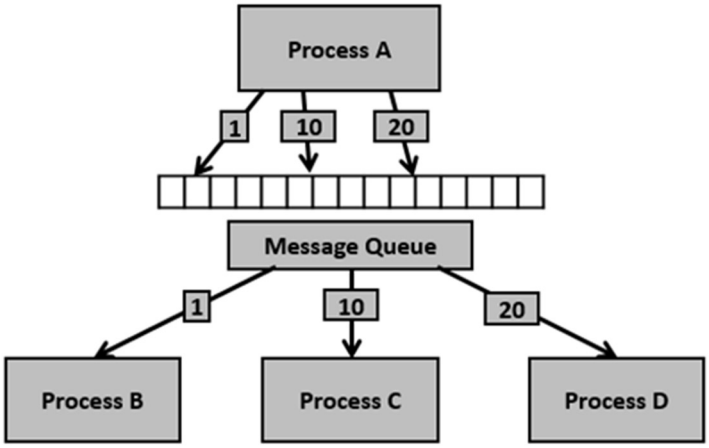
region of RAM, exclusive of operating system kernel space, is fundamental to IPC shared memory, as illustrated in the image below from a tutorial on IPC shared memory:



Henri Casanova, *Inter-Process Communications (IPC)*, presentation for class ICS332—Operating Systems, University of Hawaii (2018), *available at* http://www2.hawaii.edu/~esb/2018fall.ics332/sep17.pdf. Each of these processes having a message list that is a message queue is also fundamental to IPC shared memory, as illustrated in the image below from another tutorial on IPC shared memory:



- Writing into the shared memory by one process with different data packets and reading from it by multiple processes, i.e., as per message type.



Having seen certain information on message queues, now it is time to check for the system call (System V) which supports the message queues.

To perform communication using message queues, following are the steps –

Step 1 – Create a message queue or connect to an already existing message queue (msgget())

Step 2 – Write into message queue (msgsnd())

Step 3 – Read from the message queue (msgrcv())

Step 4 – Perform control operations on the message queue (msgctl())

Now, let us check the syntax and certain information on the above calls.

Tutorials Point, *Message Queues*, *available at* https://www.tutorialspoint.com/inter_process_communication/inter_process_communication_message_queues.htm.

1(e): accumulating message data from said first process in a location in said message buffer;—On information and belief, Cisco makes, uses, sells, and/or offers to sell a method of accumulating message data from the first process in a location in the message buffer.

For instance, Cisco generally discloses that "[t]he operating system provides different synchronization mechanisms between processes that are writing to or reading

from shared memory regions." Mobeen Tahir et al., *Cisco IOS XR Fundamentals* (Cisco Press, 2009), p. 8. In particular, Cisco discloses that "IOS XR processes are designed to be restartable," such that "[w]hen a process crashes it can be restarted," and "[a] running process saves its vital states to a checkpoint server, which is essentially a shared memory store." *Id.* at p. 51.

In addition, accumulating message data from a first process in the message buffer is fundamental to IPC shared memory, as illustrated in the above images from tutorials on IPC shared memory. *See* above tutorial images for element 1(d) of Claim 1 of the '231 Patent.

1(f): said first process adding to said message list of said second process a memory offset corresponding to said location in said message buffer; and,—On information and belief, as part of implementing IPC shared memory, Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS XR software to perform the function of a first process adding, to the message list of a second process, a memory offset corresponding to the location in the message buffer.

For instance, IPC mechanisms that use shared memory generally involve a memory offset that may be specified for a first process ("said first process adding to said message list . . . a memory offset corresponding to said location in said message buffer"), and then that memory offset may be used by the second process to access the shared memory ("said message list of said second process"). As explained in documentation for modern IPC shared memory:

> Once created or opened, a process just has to map the shared memory object in the process' address space. The user can map the whole shared memory or just part of it. The mapping process is done using the mapped_region class. The class represents a memory region that has been mapped from a

> shared memory or from other devices ***that have also mapping capabilities*** (for example, files). A mapped_region can be created from any memory_mappable object and as you might imagine, shared_memory_object is a memory_mappable object . . . The user can ***specify the offset*** from the mappable object where the mapped region should start and the size of the mapped region. If no offset or size is specified, the whole mappable object (in this case, shared memory) is mapped. If the offset is specified, but not the size, the mapped region covers from the offset until the end of the mappable object.

Boost C++ Libraries, *Sharing Memory Between Processes*, *available at* https://www.boost.org/doc/libs/1_47_0/doc/html/interprocess/sharedmemorybetween processes.html (emphasis added).

1(g): manipulating in said second process said accumulated data at said location corresponding to said offset,—On information and belief, as part of implementing IPC shared memory, Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS XR software to perform the function of manipulating in the second process the accumulated data at the location corresponding to the offset.

For instance, IPC mechanisms that use shared memory generally involve a second process that manipulates data accumulated at a shared memory, the location of which can be an offset specified by a first process. *See, e.g.,* Boost C++ Libraries, *Sharing Memory Between Processes*, especially the above block quote in element 1(f) of Claim 1 of the '231 Patent.

1(h): whereby said accumulated message data is transferred from said first process to said second process with minimal data transfer overhead.—On information and belief, Cisco makes, uses, sells, and/or offers to sell a device that operates with the Cisco IOS XR software to perform a method wherein accumulated message data is transferred from the first process to the second process with minimal data transfer overhead.

For instance, "Cisco IOS XR supports the concept of a two-stage commit model" that helps apply a target configuration "in bulk," which "in turn helps in doing subsequent operations such as verification, checkpointing, logging, and so on in bulk." Mobeen Tahir et al., *Cisco IOS XR Fundamentals* (Cisco Press, 2009), p. 110. Specifically, Cisco discloses that "[b]ulk operation is especially good for performance because it reduces interprocess communication (IPC) overhead." *Id*; *see also, e.g.,* https://blogs.cisco.com/performance/shared-memory-as-an-mpi-transport-part-2 ("With shared memory, MPI completely controls exactly what protocol bytes are transferred, and can avoid that unnecessary overhead.").

Moreover, reducing data transfer overhead is fundamental to IPC mechanisms that use shared memory. *See, e.g.,* Code Project, *Fast IPC Communication Using Shared Memory and InterlockedCompareExhange (Updated!)*, Sept. 28, 2006, *available at* https://www.codeproject.com/Articles/14740/Fast-IPC-Communication-Using-Shared-Memory-and-Int ("Not only is the shared memory implementation one of the easiest to implement, it's also one of the fastest. Why would this be one of the fastest, I hear you ask? ***Minimal overhead***. Overhead is generated whenever you make a call to another function. Be it a kernel or library, if your IPC makes no calls to any other function, then you've done away with a large bottleneck. Shared memory IPCs have no requirement for third party function calls.") (emphasis added); *see also* Henri Casanova, *Inter-Process Communications (IPC)*, presentation for class ICS332—Operating Systems, University of Hawaii (2018).

139. Additionally, Defendant Cisco has been and/or currently is an active inducer of infringement of the '231 Patent under 35 U.S.C. § 271(b) and contributory infringer of the

‘231 Patent under 35 U.S.C. § 271(c).

140. Cisco knew of the ‘231 Patent, or at least should have known of the ‘231 Patent, but was willfully blind to its existence. On information and belief, Cisco has had actual knowledge of the ‘231 Patent since at least as early as the filing and/or service of this Complaint.

141. Cisco has provided the Accused ‘231 Products to its customers and, on information and belief, instructions to (i) use the Accused ‘231 Products in an infringing manner and/or (ii) make an infringing device, while being on notice of (or willfully blind to) the ‘231 Patent and Cisco’s infringement. Therefore, on information and belief, Cisco knew or should have known of the ‘231 Patent and of its own infringing acts, or deliberately took steps to avoid learning of those facts.

142. Cisco knowingly and intentionally encourages and aids at least its end-user customers to directly infringe the ‘231 Patent.

143. Cisco’s end-user customers directly infringe at least one or more claims of the ‘231 Patent by using the Accused ‘231 Products in their intended manner to infringe. Cisco induces such infringement by providing the Accused ‘231 Products and instructions to enable and facilitate infringement, knowing of, or being willfully blind to the existence of, the ‘231 Patent. On information and belief, Cisco specifically intends that its actions will result in infringement of one or more claims of the ‘231 Patent, or subjectively believe that their actions will result in infringement of the ‘231 Patent, but took deliberate actions to avoid learning of those facts, as set forth above.

144. Additionally, Cisco contributorily infringes at least one or more claims of the ‘231 Patent by providing the Accused ‘231 Products and/or software components thereof, that embody a material part of the claimed inventions of the ‘231 Patent, that are known by Cisco to be

specially made or adapted for use in an infringing manner, and are not staple articles with substantial non-infringing uses. The Accused '231 Products are specially designed to infringe at least one or more claims of the '231 Patent, and their accused components have no substantial non-infringing uses. In particular, on information and belief, the software modules and code that implement and perform the infringing functionalities identified above are specially made and adapted to carry out said functionality and do not have any substantial non-infringing uses.

145. At least as early as the filing and/or service of this Complaint, Cisco's infringement of the '231 Patent was and continues to be willful and deliberate, entitling Commstech to enhanced damages.

146. Additional allegations regarding Cisco's knowledge of the '231 Patent and willful infringement will likely have evidentiary support after a reasonable opportunity for discovery.

147. Cisco's infringement of the '231 Patent is exceptional and entitles Commstech to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

148. Commstech is in compliance with any applicable marking and/or notice provisions of 35 U.S.C. § 287 with respect to the '231 Patent.

149. Commstech is entitled to recover from Cisco all damages that Commstech has sustained as a result of Cisco's infringement of the '231 Patent, including, without limitation, a reasonable royalty.

COUNT V: INFRINGEMENT OF U.S. PATENT NO. 7,769,028

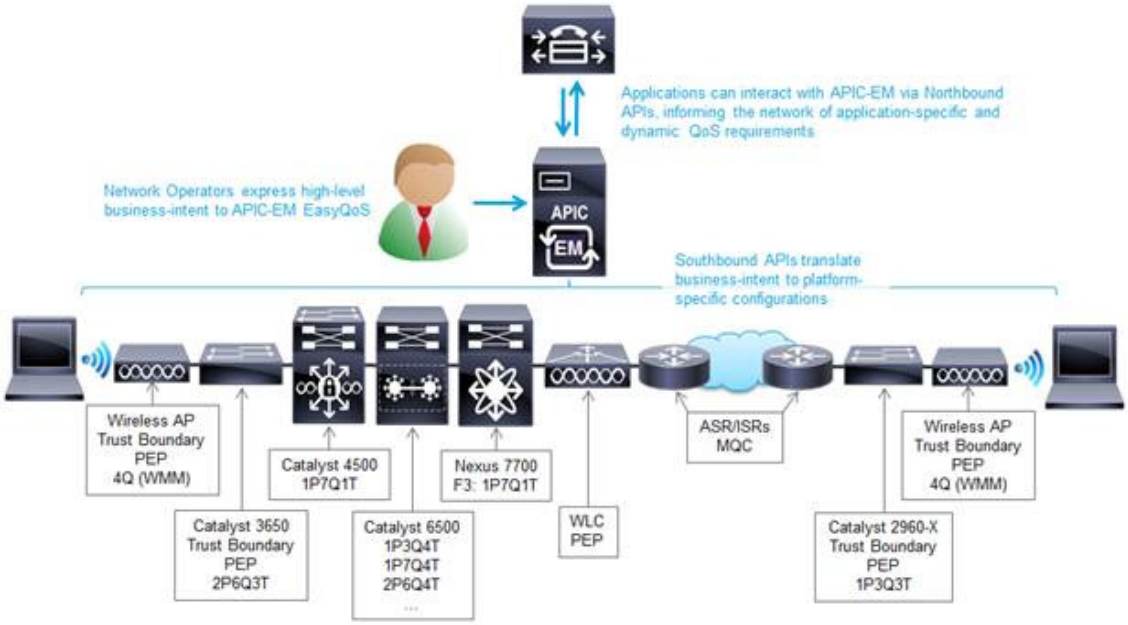
150. Commstech incorporates by reference and re-alleges paragraphs 68-80 of this Complaint as if fully set forth herein.

151. Defendant Cisco has infringed and is infringing, either literally or under the doctrine of equivalents, the '028 Patent in violation of 35 U.S.C. § 271 *et seq.*, directly and/or indirectly, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license, products that operate with the "EasyQoS" application, which supports numerous Cisco routers, switches, and/or platforms listed on Cisco's website (collectively referred to herein as the "Accused '028 Products"), that infringe at least one or more claims of the '028 Patent. *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-5-x/easyqos/supported-platforms/b_EasyQoS_Supported_Devices_1_5_x.pdf.

152. As just one non-limiting example, set forth below is exemplary evidence of infringement of Claim 13 of the '028 Patent in connection with the Accused '028 Products. This description is based on publicly available information. Commstech reserves the right to modify this description, including, for example, on the basis of information about the Accused '028 Products that it obtains during discovery.

13(a): A system for communicating data, the system including:—Cisco makes, uses, sells, and/or offers to sell a system for communicating data in accordance with Claim 13. For instance, Cisco makes, uses, sells, and/or offers to sell a computing system that has access to Cisco's "Application Policy Infrastructure Controller Enterprise Module" ("APIC-EM"), which provides a "web-based GUI" comprising several web applications, including the "EasyQoS" application that can be used to configure and deploy QoS policies to various network devices. *See* https://easyqos-16.readthedocs.io/en/latest/chapter-04.html; https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-Easy

QoS-DesignGuide-Dec2017.html#_Toc499730719; https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-1-x/config-guide/b_apic-em_config_guide_v_1-1-x/b_apic-em_config_guide_v_1-x_chapter_0111.html ("EasyQoS configures quality of service policies on devices based on the QoS feature set available on the device."). *Id.* A high-level overview of Cisco's EasyQoS solution is reproduced below:



https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.html#_Toc499730719.

13(b): a data prioritization component adapted to assign a priority to data, wherein the prioritization occurs at least one of: in a transport layer of a network communications protocol stack of a data communication system, and at a top of the transport layer of the network communications protocol stack of the data communication system;—Cisco makes, uses, sells, and/or offers to sell a system that comprises a data prioritization component adapted to assign a priority to data, where the prioritization occurs either in a

transport layer of a network communications protocol stack of a data communication system or at a top of the transport layer of the network communications protocol stack of the data communication system.

For instance, a computing system that has access to Cisco's APIC-EM includes a data prioritization component that is configured to assign priority to data in either a transport layer or at a top of the transport layer. In particular, Cisco touts that its EasyQoS application applies QoS to "prioritize applications across [a] network in minutes." *See* https://www.cisco.com/c/en_in/products/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/index.html#~stickynav=2. Indeed, Cisco's EasyQoS application supports "Next Generation Network-Based Application Recognition" (or "NBAR2") to prioritize data by assigning a priority to the data either in a transport layer or at a top of the transport layer. *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.html#_Toc499730720; https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-6-x/easyqos/user-guide/b_Cisco_EasyQoS_User_Guide_1_6_0_x.pdf ("EasyQoS supports all of the applications in the Cisco Next Generation Network-Based Application Recognition (NBAR2) library.").

According to Cisco, "[t]he NBAR 2 classification engine recognizes and classifies a wide variety of protocols and applications, including web-based and other difficult-to-classify applications and protocols that use dynamic TCP/UDP port assignments," which are two well-known protocols at the transport layer. *See* https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/network-Accbased-application-recognition-nbar/

qa_c67-697963.html; https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/network-based-application-recognition-nbar/product_bulletin_c25-627831.html. Cisco explains that "[w]hen NBAR2 recognizes and classifies a protocol or application, the network can be configured to apply the appropriate quality of service (QoS) for that application or traffic with that protocol." *See, e.g., id.*

Moreover, Cisco explains that "[t]he EasyQoS feature provides three levels of business-relevance groupings" that "essentially map to three types of traffic: high priority [(e.g., "voice, video, streaming and collaborative multimedia applications," etc.)], neutral [(e.g., "legacy applications")], and low priority [(e.g., ""consumer- and/or entertainment-oriented" applications)]. *See* https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-6-x/easyqos/user-guide/b_Cisco_EasyQoS_User_Guide_1_6_0_x.pdf.

13(c): a network analysis component adapted to determine a status of a network;—Cisco makes, uses, sells, and/or offers to sell a system that comprises a network analysis component adapted to determine a status of a network.

For instance, a computing system that has access to Cisco's APIC-EM includes a network analysis component that is configured to determine a status of a network. In particular, Cisco's EasyQoS application includes a "monitoring feature" that provides the ability to "monitor[ ] the health of WAN-connected interfaces on routers to which EasyQoS policy has been applied." *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.html. Cisco explains that "[a]pplication health can be displayed for all traffic-classes, a single traffic class, or groups of traffic-classes (Data, Video, or Control)," and an "application health score consists of

both a 'grade'—Excellent, Good, Fair, Poor, Bad, or Critical—which is based upon configurable drop thresholds, and a value from 0.0-10.0." *Id.*

13(d): a mode selection component adapted to select at least one mode based at least in part on the status of the network; and;—Cisco makes, uses, sells, and/or offers to sell a system that comprises a mode selection component adapted to select at least one mode based at least in part on the status of the network.

For instance, a computing system that has access to Cisco's APIC-EM includes a mode selection component that is configured to select at least one mode based at least in part on the status of the network. In particular, Cisco's EasyQoS application supports "dynamic QoS," which is "used on LAN interfaces [that] need a specific class of service to be in effect for the duration of some event." *See* https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-6-x/easyqos/user-guide/b_Cisco_EasyQoS_User_Guide_1_6_0_x.pdf. To illustrate an example, Cisco explains that "[d]ynamic QoS policies are used primarily in business applications, such as voice and video applications." *Id.* Cisco further explains that when a video or voice call is made (which may utilize a certain amount of bandwidth), "the call proceeds while the Cisco APIC-EM applies the QoS policies [for the video or voice traffic flow] to the relevant LAN access interfaces on which hosts (such as, IP phones or telepresence end-points) are connected," and "[w]hen the call is over, "CUCM [Cisco Unified Call Manager] signals the APIC-EM to remove the QoS policies" for the video or voice traffic flow. *Id; see also* https://video.cisco.com/detail/videos/enterprise-networks/video/4774660139001/apic-em-easy-qos-demo?autoStart=true (Cisco video demonstration showing the Cisco APIC-EM applying QoS policies when dynamic QoS is

enabled and a call is made and removing the QoS policies when the call is over).

13(e): a data communications component adapted to communicate the data based at least in part on the priority of the data and the status of the network, the data prioritization component being adapted to assign priority to the data based on prioritization rules, wherein the prioritization rules are selected based upon the selected at least one mode, wherein the data is communicated at a transmission rate metered based at least in part on the status of the network.—Cisco makes, uses, sells, and/or offers to sell a system that comprises a data communications component adapted to communicate the data based at least in part on the priority of the data and the status of the network, where the data prioritization component is adapted to assign priority to the data based on prioritization rules that are selected based upon the selected at least one mode, wherein the data is communicated at a transmission rate metered based at least in part on the status of the network.

For instance, a computing system that has access to Cisco's APIC-EM includes a data communication component that is configured to perform the communicate the data based at least in part on the priority of the data and the status of the network. Indeed, as noted above, Cisco explains that when a video or voice call is made (which may utilize a certain amount of bandwidth), "the call proceeds while the Cisco APIC-EM applies the QoS policies [for the video or voice traffic flow] to the relevant LAN access interfaces on which hosts (such as, IP phones or telepresence end-points) are connected" and "[w]hen the call is over, "CUCM [Cisco Unified Call Manager] signals the APIC-EM to remove the QoS policies" for the video or voice traffic flow. *Id; see also* https://video.cisco.com/detail/videos/enterprise-networks/video/4774660139001/apic-em-

easy-qos-demo?autoStart=true (Cisco video demonstration showing the Cisco APIC-EM applying QoS policies when dynamic QoS is enabled and a call is made and removing the QoS policies when the call is over).

Further, Cisco's EasyQoS application includes "Queuing Profiles" to customize "[t]he amount of bandwidth allocated for each of the 12 traffic-classes provisioned by EasyQoS," which include, for example, voice and broadcast video." *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.pdf. As one example to illustrate, Cisco provides a mapping of the traffic-classes and bandwidth allocations from a "default" EasyQoS queuing profile, which is reproduced in part below:

Table 11 Default Queuing Profile Mapping to 2P6Q3T Egress Queuing Policy

Traffic Class	DSCP Marking	BW % in the Default Queuing Profile	BWR % Calculated from the Default Queuing Profile	2P6Q3T Egress Queue Mapping	BWR % Allocation in 2P6Q3T Egress Queue
Voice	EF	10%	N/A	Voice-PQ1	Voice-PQ1 bandwidth is constrained to 10%, and consists of traffic from the Voice traffic-class.
Broadcast Video	CS5	10%	N/A	Video-PQ2	Video-PQ2 bandwidth is constrained to 33% and consists of traffic from the Broadcast Video, Real-Time Interactive, and Multimedia Conferencing traffic-classes.
Real-Time Interactive	CS4	13%	N/A	Video-PQ2	
Multimedia Conferencing	AF41	10%	N/A	Video-PQ2	

See id.

As shown, the default queuing profile includes a "voice" traffic-class with "bandwidth [that] is constrained to 10%," and "Broadcast Video," "Real-Time Interactive," and "Multimedia Conferencing" traffic-classes with "bandwidth [that] is constrained to 33%." *See id.*

153. Additionally, Defendant Cisco has been and/or currently is an active inducer of infringement of the '028 Patent under 35 U.S.C. § 271(b) and contributory infringer of the '028 Patent under 35 U.S.C. § 271(c).

154. Cisco knew of the '028 Patent, or at least should have known of the '028 Patent, but was willfully blind to its existence. On information and belief, Cisco has had actual knowledge of the '028 Patent since at least as early as the filing and/or service of this Complaint.

155. Cisco has provided the Accused '028 Products to its customers and, on information and belief, instructions to (i) use the Accused '028 Products in an infringing manner and/or (ii) make an infringing device, while being on notice of (or willfully blind to) the '028 Patent and Cisco's infringement. Therefore, on information and belief, Cisco knew or should have known of the '028 Patent and of its own infringing acts, or deliberately took steps to avoid learning of those facts.

156. Cisco knowingly and intentionally encourages and aids at least its end-user customers to directly infringe the '028 Patent.

157. Cisco's end-user customers directly infringe at least one or more claims of the '028 Patent by using the Accused '028 Products in their intended manner to infringe. Cisco induces such infringement by providing the Accused '028 Products and instructions to enable and facilitate infringement, knowing of, or being willfully blind to the existence of, the '028 Patent. On information and belief, Cisco specifically intends that its actions will result in infringement of one or more claims of the '028 Patent, or subjectively believe that their actions will result in infringement of the '028 Patent, but took deliberate actions to avoid learning of those facts, as set forth above.

158. Additionally, Cisco contributorily infringes at least one or more claims of the '028 Patent

by providing the Accused '028 Products and/or software components thereof, that embody a material part of the claimed inventions of the '028 Patent, that are known by Cisco to be specially made or adapted for use in an infringing manner, and are not staple articles with substantial non-infringing uses. The Accused '028 Products are specially designed to infringe at least one or more claims of the '028 Patent, and their accused components have no substantial non-infringing uses. In particular, on information and belief, the software modules and code that implement and perform the infringing functionalities identified above are specially made and adapted to carry out said functionality and do not have any substantial non-infringing uses.

159. At least as early as the filing and/or service of this Complaint, Cisco's infringement of the '028 Patent was and continues to be willful and deliberate, entitling Commstech to enhanced damages.

160. Additional allegations regarding Cisco's knowledge of the '028 Patent and willful infringement will likely have evidentiary support after a reasonable opportunity for discovery.

161. Cisco's infringement of the '028 Patent is exceptional and entitles Commstech to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

162. Commstech is in compliance with any applicable marking and/or notice provisions of 35 U.S.C. § 287 with respect to the '028 Patent.

163. Commstech is entitled to recover from Cisco all damages that Commstech has sustained as a result of Cisco's infringement of the '028 Patent, including, without limitation, a reasonable royalty.

COUNT VI: INFRINGEMENT OF U.S. PATENT NO. 7,990,860

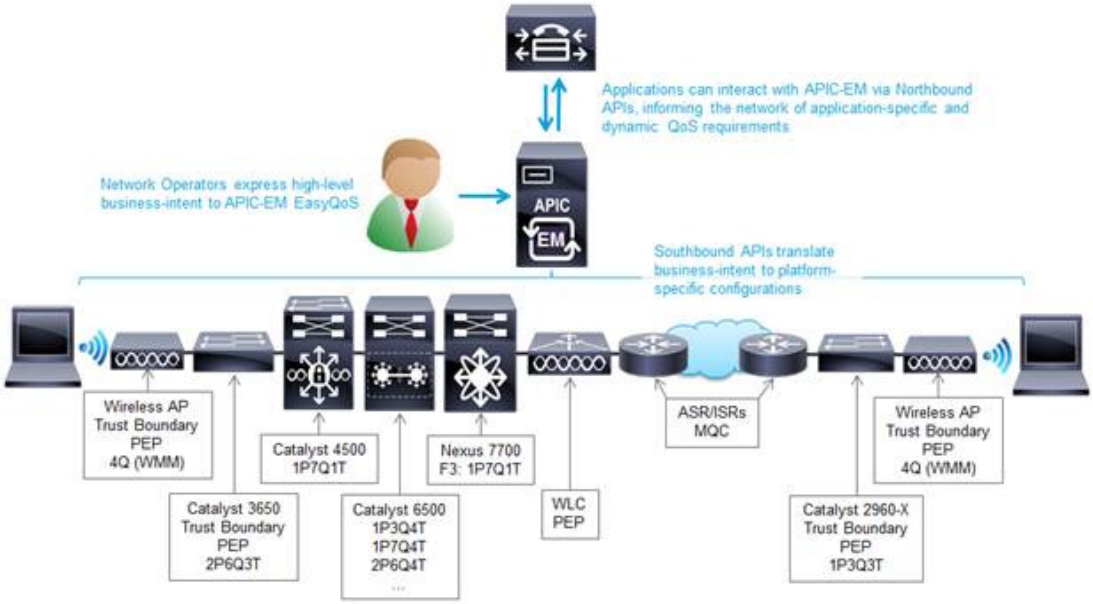
164. Commstech incorporates by reference and re-alleges paragraphs 81-93 of this Complaint as if fully set forth herein.

165. Defendant Cisco has infringed and is infringing, either literally or under the doctrine of equivalents, the '860 Patent in violation of 35 U.S.C. § 271 *et seq.*, directly and/or indirectly, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license, products that operate with the "EasyQoS" application, which supports numerous Cisco routers, switches, and/or platforms listed on Cisco's website (collectively referred to herein as the "Accused '860 Products"), that infringe at least one or more claims of the '860 Patent. *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-5-x/easyqos/supported-platforms/b_EasyQoS_Supported_Devices_1_5_x.pdf.

166. As just one non-limiting example, set forth below (with claim language in bold and italics) is exemplary evidence of infringement of Claim 15 of the '860 Patent in connection with the Accused '860 Products. This description is based on publicly available information. Commstech reserves the right to modify this description, including, for example, on the basis of information about the Accused '860 Products that it obtains during discovery.

15(a): A processing device for communicating data, the processing device including:— Cisco makes, uses, sells, and/or offers to sell a processing device for communicating data in accordance with Claim 15. For example, Cisco provides an "Application Policy Infrastructure Controller Enterprise Module" ("APIC-EM") with Cisco's "EasyQoS" application "running on top of it." *See* https://www.cisco.com/c/en/us/td/docs/

solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.html#_Toc499730719; https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-1-x/config-guide/b_apic-em_config_guide_v_1-1-x/b_apic-em_config_guide_v_1-x_chapter_0111.html. Cisco explains that "EasyQoS configures quality of service policies on devices based on the QoS feature set available on the device." *Id.* A high-level overview of Cisco's EasyQos solution is reproduced below:



See https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.html#_Toc499730719.

15(b): a network analysis component of the processing device configured to: determine a network status from a plurality of network statuses based on analysis of network measurements, and—Cisco makes, uses, sells, and/or offers to sell a processing device that comprises a network analysis component configured to determine a network status from a plurality of network statuses based on analysis of network measurements.

For instance, a Cisco device (e.g., router, switch, etc.) configured with the EasyQoS application includes a network analysis component that is programmed with the capability to "monitor[ ] the health of WAN-connected interfaces." *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.html. Cisco explains that "[a]pplication health can be displayed for all traffic-classes, a single traffic class, or groups of traffic-classes (Data, Video, or Control)," and an "application health score consists of both a 'grade'—Excellent, Good, Fair, Poor, Bad, or Critical—which is based upon configurable drop thresholds, and a value from 0.0-10.0." *Id.*

15(c): a network analysis component of the processing device configured to: determine at least one of an effective link speed and a link proportion for at least one link;—Cisco makes, uses, sells, and/or offers to sell a processing device that comprises a network analysis component configured to determine at least one of an effective link speed and a link proportion for at least one link.

For instance, a Cisco device (e.g., router, switch, etc.) configured with Cisco's EasyQoS application includes a network analysis component that is programmed with the capability to allocate bandwidth for each of the traffic-classes by "highlighting the link speed (100 Gbps, 10/40 Gbps, 1 Gbps, 100 Mbps, 10 Mbps, or 1 Mbps) and adjusting the bandwidth allocations for each traffic-class." https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.html; *see also, e.g.,* https://easyqos-16.readthedocs.io/en/latest/chapter-09.html?highlight=guarantee ("different bandwidth allocations can be configured for each of the traffic-classes based on the interface speed—1 Mbps, 10 Mbps, 100 Mbps, 1 Gbps, 10/40 Gbps, and 100 Gbps.").

To illustrate one example, Cisco explains that "the network operator can set bandwidth allocations for 1 Gbps access ports (ports connected to end-user devices) differently from 10 Gbps uplink ports (ports connected to other network infrastructure devices) within the same custom Queuing Profile." *Id.*

Further, Cisco explains that an "explicit policer" for the "Voice traffic class," which is mapped to a "low-latency queue (LLQ)," "ensures that the LLQ can use no more than the percentage of the bandwidth of the WAN link allocated to the Voice traffic class, regardless of whether there is available bandwidth." *Id.*

15(d): a mode selection component of the processing device configured to select a mode from a plurality of modes based on the determined network status, wherein each of the plurality of modes corresponds with at least one of the plurality of network statuses, wherein each of the plurality of modes comprises a user defined sequencing rule,—Cisco makes, uses, sells, and/or offers to sell a processing device that comprises a mode selection component configured to select a mode from a plurality of modes based on the determined network status, where each of the plurality of modes corresponds with at least one of the plurality of network statuses, and where each of the plurality of modes comprises a user defined sequencing rule.

For instance, Cisco's EasyQoS application supports "dynamic QoS," which is "used on LAN interfaces [that] need a specific class of service to be in effect for the duration of some event." *See* https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-6-x/easyqos/user-guide/b_Cisco_EasyQoS_User_Guide_1_6_0_x.pdf. To illustrate an example, Cisco explains that "[d]ynamic QoS policies are used primarily in business

applications, such as voice and video applications." *Id.* Cisco further explains that when a video or voice call is made (which may utilize a certain amount of bandwidth), "the call proceeds while the Cisco APIC-EM applies the QoS policies [for the video or voice traffic flow] to the relevant LAN access interfaces on which hosts (such as, IP phones or telepresence end-points) are connected," and "[w]hen the call is over, "CUCM [Cisco Unified Call Manager] signals the APIC-EM to remove the QoS policies" for the video or voice traffic flow. *Id; see also* https://video.cisco.com/detail/videos/enterprise-networks/video/4774660139001/apic-em-easy-qos-demo?autoStart=true (Cisco video demonstration showing the Cisco APIC-EM applying QoS policies when dynamic QoS is enabled and a call is made and removing the QoS policies when the call is over).

According to Cisco, the EasyQoS application allows a user with "administrator" or "policy administrator" permission to "create or change QoS policy for a group of devices that have the same policy scope." *See* https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-6-x/easyqos/user-guide/b_Cisco_EasyQoS_User_Guide_1_6_0_x.pdf.

15(e): a data prioritization component of the processing device configured to prioritize data by assigning a priority to the data, wherein the prioritization component includes a sequencing component configured to sequence the data based at least in part on the user defined sequencing rule of the selected mode;—Cisco makes, uses, sells, and/or offers to sell a processing device that comprises a data prioritization component configured to prioritize data by assigning a priority to the data, where the prioritization component includes a sequencing component configured to sequence the data based at least in part on the user defined sequencing rule of the selected mode.

For instance, Cisco touts that its EasyQoS application applies QoS to "prioritize applications across [a] network in minutes." *See* https://www.cisco.com/c/en_in/products/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/index.html#~stickynav=2. Indeed, Cisco's EasyQoS application supports "Next Generation Network-Based Application Recognition" (or "NBAR2") to prioritize data by assigning a priority to the data either in a transport layer or at a top of the transport layer. *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.html#_Toc499730720; https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-6-x/easyqos/user-guide/b_Cisco_EasyQoS_User_Guide_1_6_0_x.pdf ("EasyQoS supports all of the applications in the Cisco Next Generation Network-Based Application Recognition (NBAR2) library.").

Moreover, Cisco explains that "[t]he EasyQoS feature provides three levels of business-relevance groupings" that "essentially map to three types of traffic: high priority [(e.g., "voice, video, streaming and collaborative multimedia applications," etc.)], neutral [(e.g., "legacy applications")], and low priority [(e.g., ""consumer- and/or entertainment-oriented" applications)]. *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-6-x/easyqos/user-guide/b_Cisco_EasyQoS_User_Guide_1_6_0_x.pdf.

Further, Cisco explains that "[t]o ensure that QoS policies for the most important applications get configured on these devices, EasyQoS allocates TCAM [Ternary Content Addressable Memory] space based on . . . "1. Rank," "2. Traffic Class," "3. Popularity," and "4. Alphabetization." *See id.*

15(f): a data metering component of the processing device configured to: meter inbound data by shaping the inbound data for the at least one link, and meter outbound data by policing the outbound data for the at least one link; and —Cisco makes, uses, sells, and/or offers to sell a processing device that comprises a data metering component configured to meter inbound data by shaping the inbound data for the at least one link, and meter outbound data by policing the outbound data for the at least one link.

For instance, Cisco provides a list of devices that comprises a data metering component configured to meter inbound data by "shaping" the inbound data for the at least one link. *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/application-policy-infrastructure-controller-enterprise-module/1-4-x/easyqos/supported-platforms/b_EasyQoS_Supported_Devices_1_4_0_x.html (providing a table that lists EasyQoS features, including a "shaping" feature, that are supported by various Cisco devices). According to Cisco, "[a] shaper typically delays excess traffic using a buffer, or queueing mechanism, to hold packets and shape the flow when the data rate of the source is higher than expected." https://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c/qcfpolsh.html.

Further, Cisco explains that "[f]or traffic-classes which implement priority queuing, the policer configuration is used to determine conformed packets and exceeded packets for each collection interval," and "[p]ackets which exceed the policer are configured to be dropped by EasyQoS." *See* https://easyqos-16.readthedocs.io/en/latest/chapter-05.html.

15(g): a data communication component of the processing device configured to communicate the data based at least in part on at least one of: the priority of the data,

the effective link speed, and the link proportion.—Cisco makes, uses, sells, and/or offers to sell a processing device that comprises a data communication component configured to communicate the data based at least in part on at least one of: the priority of the data, the effective link speed, and the link proportion.

For instance, as noted above, Cisco explains that when a video or voice call is made (which may utilize a certain amount of bandwidth), "the call proceeds while the Cisco APIC-EM applies the QoS policies [for the video or voice traffic flow] to the relevant LAN access interfaces on which hosts (such as, IP phones or telepresence end-points) are connected" and "[w]hen the call is over, "CUCM [Cisco Unified Call Manager] signals the APIC-EM to remove the QoS policies" for the video or voice traffic flow. *Id; see also* https://video.cisco.com/detail/videos/enterprise-networks/video/4774660139001/apic-em-easy-qos-demo?autoStart=true (Cisco video demonstration showing the Cisco APIC-EM applying QoS policies when dynamic QoS is enabled and a call is made and removing the QoS policies when the call is over)..

Further, Cisco's EasyQoS application includes "Queuing Profiles" to customize "[t]he amount of bandwidth allocated for each of the 12 traffic-classes provisioned by EasyQoS," which include, for example, voice and broadcast video." *See, e.g.,* https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Dec2017/APIC-EM-EasyQoS-DesignGuide-Dec2017.pdf. As one example to illustrate, Cisco provides a mapping of the traffic-classes and bandwidth allocations from a "default" EasyQoS queuing profile, which is reproduced in part below:

Table 11 Default Queuing Profile Mapping to 2P6Q3T Egress Queuing Policy

Traffic Class	DSCP Marking	BW % in the Default Queuing Profile	BWR % Calculated from the Default Queuing Profile	2P6Q3T Egress Queue Mapping	BWR % Allocation in 2P6Q3T Egress Queue
Voice	EF	10%	N/A	Voice-PQ1	Voice-PQ1 bandwidth is constrained to 10%, and consists of traffic from the Voice traffic-class.
Broadcast Video	CS5	10%	N/A	Video-PQ2	Video-PQ2 bandwidth is constrained to 33% and consists of traffic from the Broadcast Video, Real-Time Interactive, and Multimedia Conferencing traffic-classes.
Real-Time Interactive	CS4	13%	N/A	Video-PQ2	
Multimedia Conferencing	AF41	10%	N/A	Video-PQ2	

See id.

As shown, the default queuing profile includes a "voice" traffic-class with "bandwidth [that] is constrained to 10%," and "Broadcast Video," "Real-Time Interactive," and "Multimedia Conferencing" traffic-classes with "bandwidth [that] is constrained to 33%." *See id.*

167. Additionally, Defendant Cisco has been and/or currently is an active inducer of infringement of the '860 Patent under 35 U.S.C. § 271(b) and contributory infringer of the '860 Patent under 35 U.S.C. § 271(c).

168. Cisco knew of the '860 Patent, or at least should have known of the '860 Patent, but was willfully blind to its existence. On information and belief, Cisco has had actual knowledge of the '860 Patent since at least as early as the filing and/or service of this Complaint.

169. Cisco has provided the Accused '860 Products to its customers and, on information and belief, instructions to use the Accused '860 Products in an infringing manner while being on notice of (or willfully blind to) the '860 Patent and Cisco's infringement. Therefore, on information and belief, Cisco knew or should have known of the '860 Patent and of its own

infringing acts, or deliberately took steps to avoid learning of those facts.

170. Cisco knowingly and intentionally encourages and aids at least its end-user customers to directly infringe the '860 Patent.

171. Cisco's end-user customers directly infringe at least one or more claims of the '860 Patent by using the Accused '860 Products in their intended manner to infringe. Cisco induces such infringement by providing the Accused '860 Products and instructions to enable and facilitate infringement, knowing of, or being willfully blind to the existence of, the '860 Patent. On information and belief, Cisco specifically intends that its actions will result in infringement of at least one or more claims of the '860 Patent, or subjectively believe that their actions will result in infringement of the '860 Patent, but took deliberate actions to avoid learning of those facts, as set forth above.

172. Additionally, Cisco contributorily infringes at least one or more claims of the '860 Patent by providing the Accused '860 Products and/or software components thereof, that embody a material part of the claimed inventions of the '860 Patent, that are known by Cisco to be specially made or adapted for use in an infringing manner, and are not staple articles with substantial non-infringing uses. The Accused '860 Products are specially designed to infringe at least one or more claims of the '860 Patent, and their accused components have no substantial non-infringing uses. In particular, on information and belief, the software modules and code that implement and perform the infringing functionalities identified above are specially made and adapted to carry out said functionality and do not have any substantial non-infringing uses.

173. At least as early as the filing and/or service of this Complaint, Cisco's infringement of the '860 Patent was and continues to be willful and deliberate, entitling Commstech to

enhanced damages.

174. Additional allegations regarding Cisco's knowledge of the '860 Patent and willful infringement will likely have evidentiary support after a reasonable opportunity for discovery.

175. Cisco's infringement of the '860 Patent is exceptional and entitles Commstech to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

176. Commstech is in compliance with any applicable marking and/or notice provisions of 35 U.S.C. § 287 with respect to the '860 Patent.

177. Commstech is entitled to recover from Cisco all damages that Commstech has sustained as a result of Cisco's infringement of the '860 Patent, including, without limitation, a reasonable royalty.

PRAYER FOR RELIEF

WHEREFORE, Commstech respectfully requests:

A. That Judgment be entered that Cisco has infringed at least one or more claims of the Patents-in-Suit, directly and/or indirectly, literally and/or under the doctrine of equivalents;

B. An award of damages sufficient to compensate Commstech for Cisco's infringement under 35 U.S.C. § 284, including an enhancement of damages on account of Cisco's willful infringement;

C. That the case be found exceptional under 35 U.S.C. § 285 and that Commstech be awarded its reasonable attorneys' fees;

D. Costs and expenses in this action;

E. An award of prejudgment and post-judgment interest; and

F. Such other and further relief as the Court may deem just and proper.

DEMAND FOR JURY TRIAL

Pursuant to Rule 38(b) of the Federal Rules of Civil Procedure, Commstech respectfully demands a trial by jury on all issues triable by jury.

Respectfully submitted,

Dated: May 9, 2019

TOLER LAW GROUP PC
and
LEE SULLIVAN SHEA & SMITH LLP

By: */s/ M. Vanessah Pace*

M. Vanessah Pace, Texas Bar No. 24028818
vpace@tlgiplaw.com
TOLER LAW GROUP PC
8500 Bluffstone Cove, Suite A201
Austin, TX 78759
Tel: (512) 327-5515
Fax: (512) 327-5575

George I. Lee (pending *pro hac vice*)
Illinois ID No. 62254230, lee@ls3ip.com
Sean M. Sullivan (pending *pro hac vice*)
Illinois ID No. 6230605, sullivan@ls3ip.com
Michael P. Boyea (pending *pro hac vice*)
Illinois ID No. 6312399, boyea@ls3ip.com
Cole B. Richter (pending *pro hac vice*)
Illinois ID No. 6315686, richter@ls3ip.com
Jae Y. Pak (pending *pro hac vice*)
Illinois ID No. 6321249, pak@ls3ip.com
LEE SULLIVAN SHEA & SMITH LLP
656 West Randolph Street, Floor 5W
Chicago, IL 60661
Tel: (312) 754-9602
Fax: (312) 754-9603

Attorneys for Plaintiff
Commstech Holdings LLC